2003-03

A hypermedia representation of a taxonomy of usability characteristics in virtual environments

Tokgoz, Asim

Monterey, California. Naval Postgraduate School

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

A HYPERMEDIA REPRESENTATION OF A TAXONOMY OF USABILITY CHARACTERISTICS IN VIRTUAL ENVIRONMENTS

by

Asim Tokgoz

March 2003

Thesis Advisor:	Rudolph P. Darken
Co-Advisor:	Joseph A. Sullivan

This thesis done in cooperation with the MOVES Institute

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE	*Form Approved OMB No. 0704-0188*

1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** March 2003	**3. REPORT TYPE AND DATES COVERED** Master's Thesis

4. TITLE AND SUBTITLE Title: A Hypermedia Representation of a Taxonomy of Usability Characteristics in Virtual Environments	**5. FUNDING NUMBERS**
6. AUTHOR (S) Asim Tokgoz	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000	**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A	**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the U.S. Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited	**12b. DISTRIBUTION CODE**

13. ABSTRACT *(maximum 200 words)*

The goal of much work in Virtual Environments (VEs) to date has been to produce innovative technology but until recently, there has been very little user-centered, usability-focused research in VEs that will turn interesting applications into usable ones. There is beginning to be at least some awareness of the need for usability engineering within the VE community. A handful of articles address usability concerns for particular parts of the VE usability space. From this point Gabbard and Hix [1997] has proposed a taxonomy about usability characteristics in VEs to help VE usability engineers and designers. This taxonomy can be used to learn characteristics of VEs or to develop usability engineering methodologies specifically for VEs.

In this study, we built hypermedia representation of the taxonomy and evaluated the effectiveness of the user interface by using scenario based formative usability engineering method that developed by Hix and Hartson [1993]. First, we discussed the need for usability engineering for VEs and took a look at a proposed usability engineering methodology [Gabbard and others, 1999] for VEs. Second we implemented hypermedia based web-site taxonomy and then evaluated it iteratively. Last, we added a new study to show the dynamic nature of web-site application.

14. SUBJECT TERMS Taxonomy, Usability, Usability Characteristics, Human Factors, Human Computer Interaction, Virtual Environments, Evaluation, Formative Evaluation, Experiment, User-Centered Design, Web Site Design	**15. NUMBER OF PAGES** 162
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	**18. SECURITY CLASSIFICATION OF THIS PAGE**	**19. SECURITY CLASSIFICATION OF ABSTRACT**	**20. LIMITATION OF ABSTRACT**
Unclassified	Unclassified	Unclassified	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

This thesis done in cooperation with the MOVES Institute

Approved for public release; distribution is unlimited

A HYPERMEDIA REPRESENTATION OF A TAXONOMY OF USABILITY CHARACTERISTICS IN VIRTUAL ENVIRONMENTS

Asim Tokgoz
Lieutenant Junior Grade, Turkish Navy
B.S., Turkish Naval Academy, 1997

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN MODELING, VIRTUAL ENVIRONMENTS AND SIMULATION

from the

NAVAL POSTGRADUATE SCHOOL
March 2003

Author: Asim Tokgoz

Approved by: Rudolph P. Darken
Thesis Advisor

Joseph A. Sullivan
Co-Advisor

Rudolph P. Darken, Chair, Academic Committee
Modeling, Virtual Environments and Simulation

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The goal of much work in Virtual Environments (VEs) to date has been to produce innovative technology but until recently, there has been very little user-centered, usability-focused research in VEs that will turn interesting applications into usable ones. There is beginning to be at least some awareness of the need for usability engineering within the VE community. A handful of articles address usability concerns for particular parts of the VE usability space. From this point Gabbard and Hix [1997] has proposed a taxonomy about usability characteristics in VEs to help VE usability engineers and designers. This taxonomy can be used to learn characteristics of VEs or to develop usability engineering methodologies specifically for VEs.

In this study, we built hypermedia representation of the taxonomy and evaluated the effectiveness of the user interface by using scenario based formative usability engineering method that developed by Hix and Hartson [1993]. First, we discussed the need for usability engineering for VEs and took a look at a proposed usability engineering methodology [Gabbard and others, 1999] for VEs. Second we implemented hypermedia based web-site taxonomy and then evaluated it iteratively. Last, we added a new study to show the dynamic nature of web-site application.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF FIGURES

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS

ASP	Active Server Page
CS	Computer Science
CSS	Cascading Style Sheet
ERD	Entity Relationship Diagram
GUI	Graphical User Interface
HTML	Hyper Text Markup Language
MOVES	Modeling, Virtual Environments and Simulation
NPS	Naval Postgraduate School, Monterey, CA
MRP	Maximal Repeating Patterns
RFP	Requests for Proposal
XML	Extensible Markup Language
VE	Virtual Environments
WIMP	Windows, Icons, Menus (Mice) and Pointers
WWW	World Wide Web

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGEMENTS

I would like to express acknowledgment and appreciation to my thesis committee members, Dr. Rudy Darken and Cmdr. Joe Sullivan for their assistance, direction and dedication throughout my course of study. Acknowledgment and appreciation is also due to Dr. Deborah Hix and Joseph L. Gabbard for their cooperation and permission to use their study in this thesis.

Acknowledgment and gratitude is expressed to Arzu Tokgoz for her love, support, sacrifice, and encouragement needed to bring this study completion. Acknowledgment and gratitude is also expressed to my son Osman Sungur Tokgoz for his support by not sleeping in the nights and making noises to disturb me.

Finally, I want to thank my country for giving me the opportunity to obtain such a good education at the Naval Postgraduate School, Monterey, CA.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. OVERVIEW

The goal of much work in Virtual Environments (VEs) to date has been to produce innovative technology but until recently, there has been very little user-centered, usability-focused research in VEs that will turn interesting applications into usable ones. An underlying assumption among both researchers and developers sometimes seems to be that VEs, because they are novel, impressive, and provide *natural* interaction, are inherently good and usable. Progress is needed to move beyond this flawed assumption, to have usability engineering become a routine activity in VE development, with methods to produce VEs that are effective and efficient for their users, not merely new and different [Gabbard and Hix, 1998].

There is beginning to be at least some awareness of the need for usability engineering within the VE community. A handful of articles address usability concerns for particular parts of the VE usability space. For example, some have published guidelines for spatial input devices (e.g., [Hinckley and others, 1994]), hints for three dimensional interface design (e.g., [Bricken, 1990]), usability in learning-based VEs (e.g., [Salzman and others, 1995]), and usability issues in haptic feedback hardware (e.g., [Hannaford and Venema, 1995]). However, many publications that include usability issues fail to address the complex inter-dependencies present in VEs among users, tasks, input devices, output devices, etc. Stuart [1996], an excellent book on VE design, gives broad coverage to many of the issues that are important in design of usable VEs [Gabbard and Hix, 1998].

Existing usability methodologies, such as those for Graphical User Interfaces (GUIs) need extensive assessment and modifications to support invention, development, and study of VE user interfaces. Thus, there is a need to produce a new generation of methods specifically for usability engineering of VEs. But challenges to produce usability engineering methods for VEs include

lack of taxonomy as a structured basis for method development [Gabbard and Hix, 1998].

As a major step in creating new methods for usability engineering of VEs, Gabbard and Hix [1997] have produced a comprehensive taxonomy of usability characteristics specifically for VEs, and supplemental VE usability resources in the form of *design guidelines, context-driven discussion* and *references* [Gabbard and Hix, 1998]. This research will be our focus point.

B. BACKGROUND

In order to build user-centered VEs, designers and builders need some methodologies which can be applied to VEs. Actually there are methodologies for classic GUIs but VEs do not carry the same characteristics with GUIs. GUIs usually use Windows, Icons, Menus and Pointers (WIMP) interfaces and they are simpler then VEs. People get used to these interfaces and now they are very common.

Usability engineers are trying to improve methodologies for usability of VEs. One of these methodologies is proposed by Gabbard and others [1999]. This method will be explained in this section to show where the taxonomy falls in.

Most extant usability engineering methods widely in current use were spawned by the development of GUIs. So even when VE developers attempt to apply usability engineering methods, most VE user interfaces are so radically different that well-proven techniques that have produced usable GUIs may be neither particularly appropriate nor effective for VEs. Few principles for design of VE user interfaces exist, and almost none are empirically derived or validated. Use of usability engineering methods often results in VE designs that produce much unexpected reactions and performance of users, reaffirming the need for exactly such methods! Ultimately researchers and developers of VEs should seek to improve VE applications, *from a user's perspective* — ensuring their usability — by following a systematic approach to VE development such as offered from usability engineering methods [Gabbard and Hix, 2001].

There is some research at Virginia Tech and Virtual Prototyping and Simulation Technologies, Inc (VPST) to provide a methodology — or a set of methodologies — to ensure usable and useful VE interfaces.

To this end, Gabbard and others [1999] present several usability engineering methods, mostly adapted from GUI development, that have been successfully applied to VE development. These methods include user task analysis, expert guidelines-based evaluation (also sometimes called heuristic evaluation or usability inspection), formative usability evaluation and summative comparative evaluations. Further, they postulate that — like GUI development — there is no single method for VE usability engineering, and they address how each of these methodologies supports focused, specialized design, measurement, management, and assessment techniques.

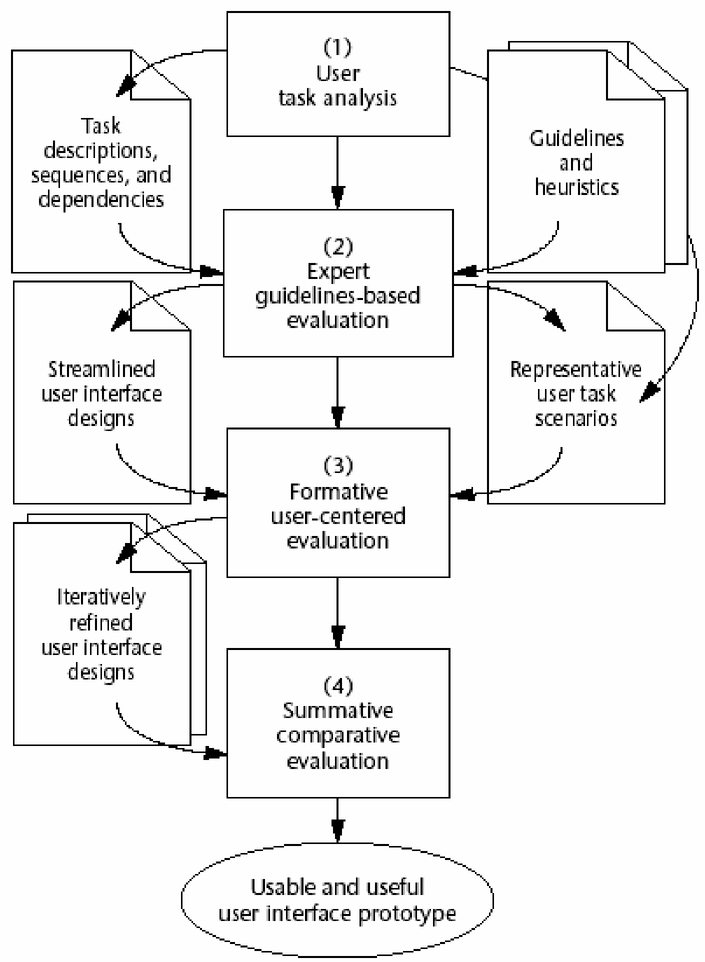


Figure 1. Methodology for the User-Centered Design and Evaluation of VE User Interaction [From Gabbard and others, 1999].

Let's take a look at the proposed methodology more closely. This methodology, illustrated in Figure 1, is based on sequentially performing [Gabbard and others, 1999]:

1. user task analysis,
2. expert guidelines-based evaluation,
3. formative user-centered evaluation, and
4. summative comparative evaluations.

Let's discuss each in more detail:

1. User Task Analysis

A *user task analysis* [Hix and Hartson, 1993; Hackos and Redish, 1998] is the process of identifying a complete description of tasks, subtasks, and methods required to use a system, as well as other resources necessary for user(s) and the system to cooperatively perform tasks. It follows a formal methodology, described in detail elsewhere [Hix and Hartson, 1993; Hackos and Redish, 1998]. As depicted in Figure 2, a user task analysis represents insights gained through an understanding of user, organization, and social workflow; needs analysis; and user modeling. A user task analysis generates critical information used throughout all stages of the application development life cycle (and subsequently, all stages of the usability design and evaluation life cycle). A

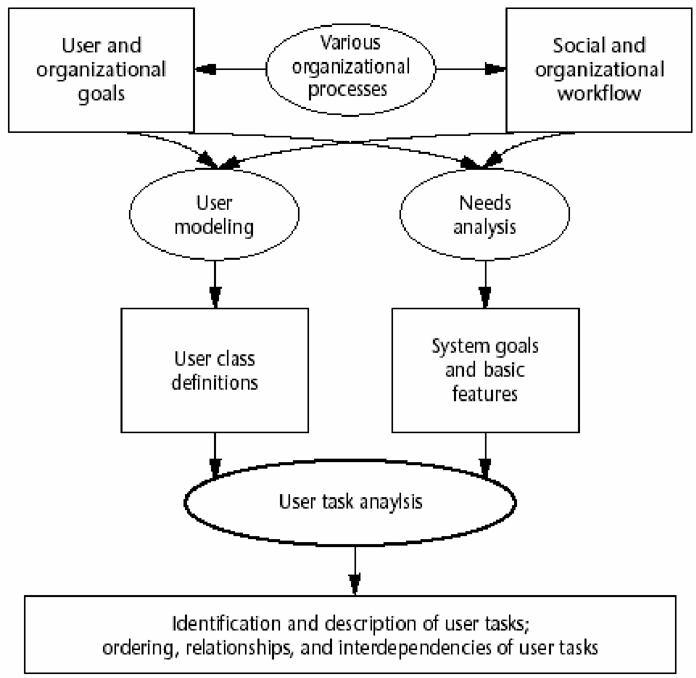


Figure 2. A User Task Analysis Identifies and Describes User Tasks as well as Their Ordering, Relationships, and Interdependencies [From Gabbard and others, 1999].

major result is a top-down decomposition of detailed user task descriptions for use by designers and evaluators. Equally revealing results include an understanding of required task sequences as well as sequence semantics. Thus, the results include not only the identification and description of tasks, but also information about the ordering, relationships, and interdependencies among user tasks [Gabbard and others, 1999].

Unfortunately, this critical step of user interaction development is often overlooked or poorly done. Without a clear understanding of user task requirement, both evaluators and developers must *best guess* or interpret desired functionality, which inevitably leads to poor user interaction design. Indeed, user interaction developers as well as user interface software developers claim that poor, incomplete, or missing user task analysis is one of the most common causes of poor user interaction design [Gabbard and others, 1999].

2. Expert Guidelines-Based Evaluation

Expert guidelines-based evaluation (heuristic evaluation or usability inspection) aims to identify potential usability problems by comparing a user interaction design —either existing or evolving— to established usability design guidelines. In this analytical evaluation, an expert in user interaction design assesses a particular interface prototype by determining what usability design guidelines it violates and supports. Then, based on these findings, especially the violations, the expert makes recommendations to improve the design. In the case of VEs, this proves particularly challenging because so few guidelines exist specific to VE user interaction [Gabbard and others, 1999].

Typically more than one person performs guidelines-based evaluations, since it's unlikely that any one person could identify all if not most of an interaction design's usability problems. Nielsen [1994] recommends three to five evaluators for a GUI heuristic evaluation, since fewer evaluators generally cannot identify enough problems to warrant the expense, while more evaluators produce diminishing results at higher costs. It's not clear whether this recommendation is

cost effective for VEs, since more complex VE interaction designs may require more evaluators than do GUIs [Gabbard and others, 1999].

Each evaluator first inspects the design independently of other evaluators' findings. Results are then combined, documented, and assessed as evaluators communicate and analyze both common and conflicting usability findings. Further, Nielsen [1994] suggests a two-pass approach. During the first pass, evaluators gain an understanding of the general flow of interaction. During the second pass, evaluators identify specific interaction components and conflicts as they relate to both task flow and the larger-scoped interaction paradigm. This method is best applied early in the development cycle so that design issues can be addressed as part of the iterative design and development process [Gabbard and others, 1999].

Expert guidelines-based evaluations rely on established usability guidelines to determine whether a user interaction design supports intuitive user task performance [Nielson, 1994; Nielson and Molich, 1990]. While these heuristics are considered the de facto standard for GUIs, they are found too general, ambiguous, and high level for effective and practical heuristic evaluation of VEs [Gabbard and others, 1999].

Recently, Gabbard and Hix [1997] produced a set of usability design guidelines specifically for VEs, contained within a taxonomy of usability characteristics. This taxonomy document provides a reasonable starting point for heuristic evaluation of VEs. The complete document contains several associated usability resources, including specific usability guidelines, detailed context-driven discussion of the numerous guidelines, and citations of additional references.

The taxonomy organizes VE user interaction design guidelines and the related context-driven discussion into four major areas:

1. users and user tasks,
2. input mechanisms,

3. virtual models, and
4. presentation mechanisms.

The taxonomy categorizes 195 guidelines covering many aspects of VEs that affect usability, including locomotion, object selection and manipulation, user goals, fidelity of imagery, input device modes and usage, interaction metaphors, and more [Gabbard and others, 1999].

The guidelines presented within the taxonomy document suit performing guidelines-based evaluation of VE user interfaces and interaction, since they provide broad coverage of VE interaction and interfaces yet are specific enough for practical application. For example, with respect to navigation within VEs, one guideline reads [Gabbard and others, 1999]:

> Provide information so that users can always answer the questions: Where am I now? What is my current attitude and orientation? Where do I want to go? How do I travel there?

Another guideline addresses methods to aid in usable object selection techniques, stating,

> Use transparency to avoid occlusion during selection.

Hypermedia representation of this taxonomy will be the objective of this study. More detailed information about the structure of this taxonomy will be presented in the following chapter (Problem Definition). As you can see, the taxonomy plays an important and vital role at this point and falls in this section.

3. Formative User-Centered Evaluation

Formative user-centered evaluation [Hix and Hartson, 1993] is a type of empirical, observational assessment *with users* that begins in earliest phases of user interaction design and continues throughout the entire life cycle. Formative evaluation produces both qualitative (narrative) and quantitative (numeric) results. The purpose of formative evaluation is to iteratively and quantifiably assess and improve the user interaction design [Hix and others, 1999].

Figure 3 shows the steps of a typical formative evaluation cycle. The cycle begins with development of user task scenarios, which are specifically designed to exploit and explore all identified task, information, and work flows. Note that user task scenarios derive from results of the user task analysis. Moreover, these scenarios should provide adequate coverage of tasks as well as accurate sequencing of tasks identified during the user task analysis. Representative users perform these tasks as evaluators collect data. These data are then analyzed to identify user interaction components or features that both support and detract from user task performance. These observations are in turn used to suggest user interaction design changes as well as formative evaluation scenario and observation (re)design [Gabbard and others, 1999].

An important point to note in the formative evaluation process is that both qualitative and quantitative data are collected from representative users during their performance of task scenarios. Developers often have the false impression that usability evaluation is something rather warm and fuzzy, with no *real* process and collecting no *real* data. Quite the contrary is true; experienced usability evaluators collect large volumes of both qualitative data and quantitative data [Hix and others, 1999; Gabbard and others, 1999].

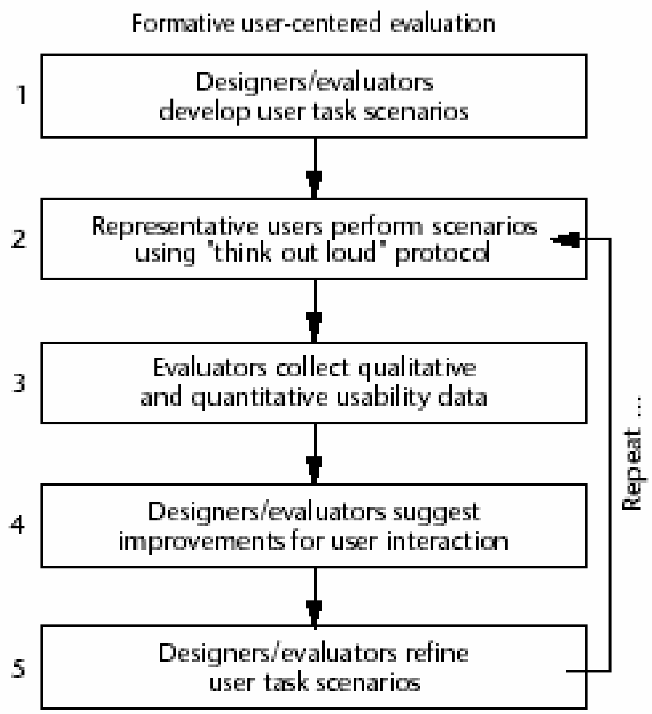


Figure 3. Formative User-Centered Evaluation Process [From Gabbard and others, 1999]

Qualitative data are typically in the form of *critical incidents* [Hix and Hartson, 1993; del Galdo and others, 1986]. A critical incident occurs while a user is performing task scenarios, and is an event that has a significant effect, either positive or negative, on user task performance or user satisfaction with the interface. Events that affect user performance or satisfaction therefore have an impact on usability. Typically a critical incident is a problem that a user encounters (e.g., an error, being unable to complete a task scenario, confusion, etc.) [Hix and others, 1999].

Quantitative data are generally related, for example, to how long it takes and the number of errors while a user is performing task scenarios. These data are then compared to appropriate baseline metrics. Quantitative data generally indicate that a problem has, occurred; qualitative data indicate where (and sometimes why) it occurred [Hix and others, 1999].

Collection of both these types of data is a key part of the formative evaluation process.

3. Summative Comparative Evaluation

Summative comparative evaluation [Hix and Hartson, 1993], in contrast to formative user-centered evaluation, is empirical assessment with users of an interaction design comparison with other interaction designs for performing the same user tasks. Summative evaluation is typically performed when there are some more-or-less *final* versions of the interaction designs, and it yields primarily quantitative results. The purpose of summative evaluation is to statistically compare user performance with different interaction designs, for example, to determine which one is better, where *better* is defined in advance [Hix and others, 1999].

When used to assess user interfaces, summative evaluation can be thought of as experimental evaluation with users comparing two or more configurations of user interface components, interaction paradigms, interaction devices, and so forth. Comparing devices and interaction techniques employs a consistent set of user task scenarios (developed during formative evaluation and

refined for summative evaluation) resulting in primarily quantitative data results that compare (on a task by task basis) the designs' ability to support user task performance [Hix and others, 1999]. (For more information see [Gabbard and Hix, 2001; Gabbard and others, 1999])

5. An Effective Progression

Gabbard and others [1999] recently did some work about user-centered VE usability and found that the progression of methods they present suits cost-effective, efficient, design and evaluation of VEs particularly well [Hix and others, 1999; Gabbard and others, 1999a]. Refer to Figure 1 throughout the following discussion.

A user task analysis provides the basis for design and evaluation in terms of what types of tasks and task sequences users will need to perform within a specific VE. This analysis generates (among other outputs) a list of detailed task descriptions, sequences, and relationships, user work, and information flow. It provides a basis for design and application of subsequent evaluation methods [Gabbard and others, 1999].

For example, the user task analysis may help eliminate or identify specific guidelines or sets of guidelines during expert guidelines-based evaluation. In a similar fashion, a user task analysis serves as both a basis for user evaluation scenario development as well as a checklist for evaluation coverage. That is, a well-developed task analysis provides evaluators with a complete list of end-use functionality detailing not only which tasks are to be performed but also likely task sequences and dependencies. Ordering and dependencies of user tasks is critical to powerful user evaluation scenario development. The closer the match between user task analysis and actual end user tasking, the better and more effective the final user interaction design [Gabbard and others, 1999]. At this point, some researchers may disagree with this idea. The match between user task analysis and actual end user tasking does not mean an effective interaction.

An expert guidelines-based evaluation is the first assessment of an interaction design based on the user task analysis and application of guidelines

for VE interaction design. This extremely useful evaluation removes many obvious usability problems from an interaction design. A VE interaction design expert will find both subtle and major usability problems through a guidelines-based evaluation. Once problems are identified, experts perform further assessment to understand how particular interaction components, devices, and so on affect user performance [Gabbard and others, 1999].

Results of expert guidelines-based evaluations are critical to effective formative and summative evaluations. For example, these results (coupled with results of user task analysis) serve as a basis for user scenario development. That is, if expert guidelines-based evaluation identifies a possible mismatch between implementation of a wireless 3D input device and manipulation of user viewpoint, then scenarios requiring users to manipulate the viewpoint should be included in formative evaluations [Gabbard and others, 1999].

Results of expert guidelines-based evaluations are also used to streamline subsequent evaluations. Further, critical usability problems identified during expert guidelines-based evaluation are corrected prior to performing formative evaluations, affording formative evaluations that don't waste time exposing those obvious usability problems addressed by the guidelines-based evaluation [Gabbard and others, 1999].

Because formative evaluation involves typical users, it most effectively uncovers issues (such as missing user tasks) that an expert performing a guidelines-based evaluation might be unaware of. A formative evaluation following a guidelines-based evaluation can focus not on major, obvious usability issues, but rather on those more subtle and more difficult to recognize issues. This becomes especially important because of the cost of VE development [Gabbard and others, 1999].

Coupling expert guidelines-based evaluations with formative user-centered evaluation helps successfully refine GUIs. Nielson [1994] recommends alternating expert guidelines-based evaluations and formative evaluation. The

rationale is that no single method can reliably identify any and all usability problems. Indeed, guidelines-based evaluation and formative evaluation complement each other, often revealing usability problems that the other may have missed [Desurvire and others, 1992].

Finally, a summative comparative evaluation following the preceding activities compares good apples to good oranges rather than comparing possibly rotten apples to good oranges. That is, summative studies comparing VEs whose interaction design has had little or no task analysis, guidelines-based evaluation, and/or formative evaluation may really be comparing one VE interaction design that is (for whatever reasons) inherently better — in terms of usability — to a different (and worse) VE interaction design. The first three methods produce a set of well-developed, iteratively refined, user interface designs. Subsequently, the designs compared in the summative study should be as usable, and comparably usable, as feasible. This means that any differences found in a summative comparison are much more likely the result of differences in the designs' basic nature rather than true differences in usability. Again, because of the cost of VE development, this confidence in results proves especially consequential [Gabbard and others, 1999].

The progression of methods is structured at a high level for application to any VE, regardless of the hardware, software, or interaction style used. Employing case-specific task analysis, guidelines, and user task scenarios facilitates broad applicability. As such, each specific method is flexible enough to support evaluation of any VE subsystem (visual, auditory, or haptic, for example) or combination thereof [Gabbard and others, 1999].

Figure 4 shows additional properties of the three types of evaluation. The solid arrows underscore the methods' application sequence. Expert guideline-based evaluation is recommended applying first, perhaps iterating several times. The least expensive evaluation to perform and very general, it can cover large portions (if not all) of the user interface. However, expert guideline-based evaluation isn’t very precise: it gives only general indications of what might be

wrong and doesn't address how to fix usability problems [Gabbard and others, 1999].

Formative usability evaluation is applied next, which is more expensive (it requires users and task scenarios) and less general (a smaller portion of the user interface can be covered per session). However, the results are more precise, often revealing where problems occur and suggesting ways to fix them. Typically iterated several times, formative usability evaluation may lead to additional expert guidelines-based evaluation of modified or missed portions of the user interface [Gabbard and others, 1999].

Finally, summative evaluations are very expensive (requiring many more subjects than formative usability evaluations) and also extremely specific — they can answer only very narrowly defined questions. However, summative evaluations answer these questions with a high degree of precision: it's the only type of evaluation that can statistically quantify how much better one design is than another [Gabbard and others, 1999].

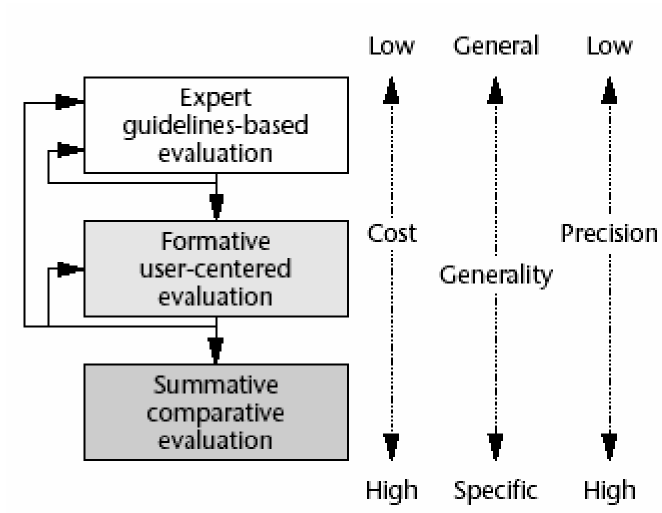


Figure 4. Additional Properties Of The Expert Guidelines-Based, Formative User-Centered, and Summative Comparative Evaluation Methods [From Gabbard and others, 1999].

The reader can get a detailed knowledge on how Gabbard and others [1999] applied their proposed methodology to some applications such as dragon battlefield visualization VE [Gabbard and Hix, 2001; Gabbard and others, 1999; Hix and others, 1999] and crumbs — a tracking tool for biological imaging

[Gabbard and Hix, 2001; Gabbard and others, 1999; Gabbard and others, 1999a].

C. PROBLEM DEFINITION

As discussed in previous section, Gabbard and Hix [1997] had developed a taxonomy to support VE designers/builders. We will convert this study to a dynamic web-based application by using iterative formative usability evaluation. In this section, the structure of this taxonomy will be explained in detail.

Gabbard and Hix [1997] structured the complete taxonomy and supplemental usability resources to support progressive disclosure, meaningful organization, and non-linear access of their comprehensive collection of VE usability resources. In particular, the taxonomy and usability resources include VE usability characteristics, specific VE usability design guidelines, context-driven discussion, and references. Access to these resources is provided through the following levels of detail [Gabbard and Hix, 1998]:

- Taxonomy of VE usability characteristics (diagram — see Figure 5)
- Specific usability design guidelines (tables — see Table 1)
- Context-driven discussion (prose)
- Reference list (alphabetized list)

1. A Taxonomy of VE Usability Characteristics

The taxonomy of VE usability characteristics is first presented in an abstract hierarchical structure represented by the four shaded boxes and their connections shown in Figure 5. This diagram depicts high-level relationships among the taxonomy's four major areas of usability issues:

- VE Users and User Tasks — general user and task characteristics and types of tasks in VEs
- VE User Interface Input Mechanisms — usability characteristics of VE input devices

- The Virtual Model — usability characteristics of generic components typically found in VEs
- VE User Interface Presentation Components — usability characteristics of VE output devices

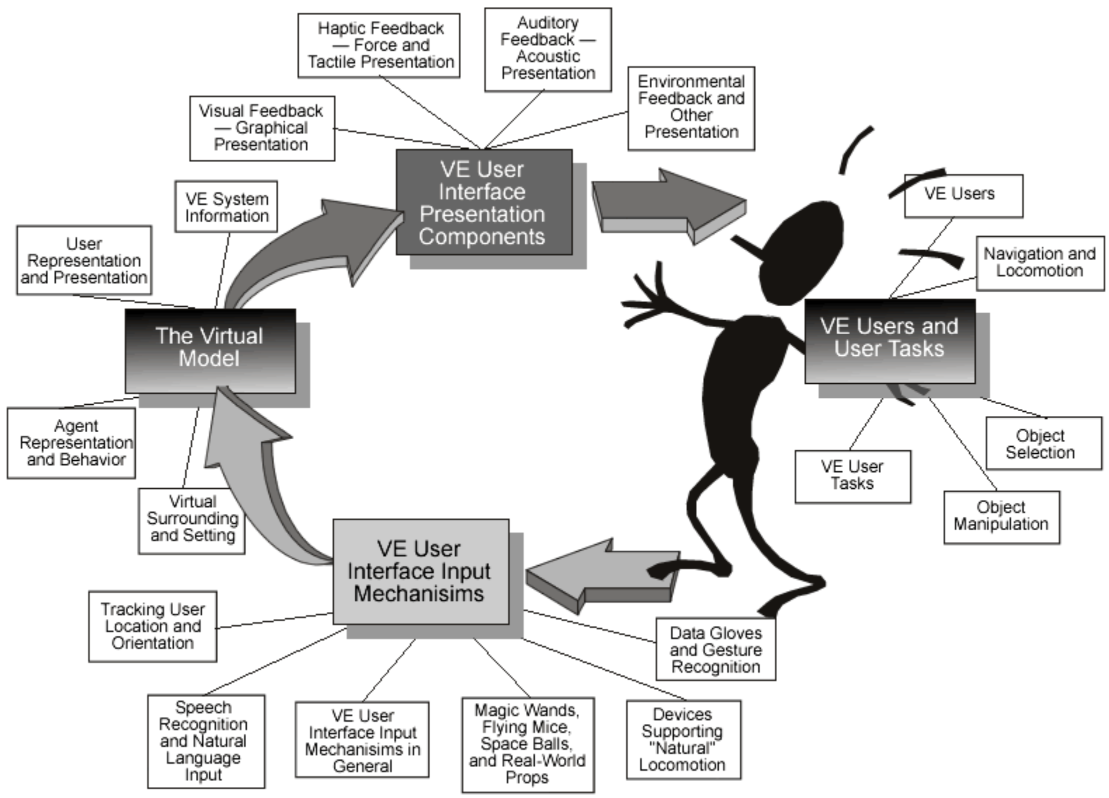


Figure 5. Overview of Taxonomy Areas [From Gabbard and Hix, 1997]

Figure 5 also contains another level of taxonomy refinement for each of these major areas, shown as white boxes. For example, *VE User Interface Presentation Components* is refined into *Visual Feedback*, *Haptic Feedback*, *Aural Feedback*, and *Environmental Feedback and Other Presentations*.

Structuring the taxonomy such that VE usability characteristics, guidelines, and research findings could be meaningfully clustered and inserted was one of author's biggest challenges. Indeed, the space of usability characteristics in VEs does not fit into a single *natural* or *correct* organization or ordering. However, some ordering had to be imposed, revealing and restricting relationships as

dictated by that particular structure. One approach to ordering a space of VE usability-related information is to use general theories of human-computer interaction as a guide. After reviewing several theories and models, they found Norman's theory of action [Norman and Draper, 1986] to be an appropriate foundation upon which to base their current organization. This theory of action defines several stages of activity and associated interdependencies that are inherent in interaction between human and machine [Norman and Draper, 1986]. It consists of several stages of user activities involved in a user's performance of a task, each of which are relevant in VE user interaction. Moreover, the theory of action is particularly well-suited for addressing how individual usability issues fit into a more abstract, larger-scale understanding of interaction between users and VEs [Gabbard and Hix, 1998].

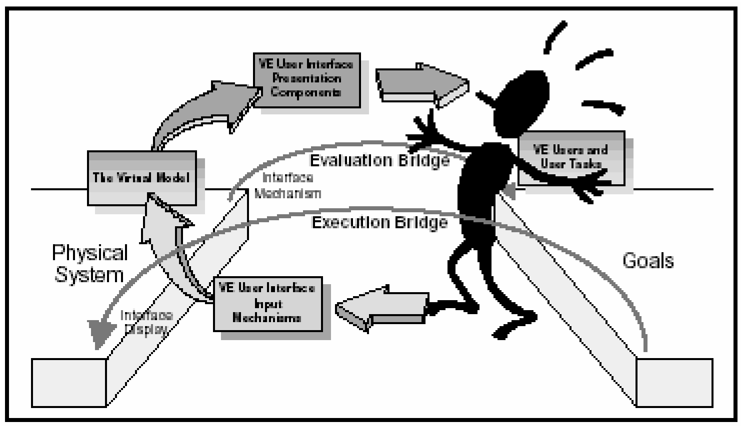


Figure 6. Structuring the Taxonomy According to Norman's Theory of Action [From Gabbard and Hix, 1998].

In particular, Norman defines a *gulf of execution*, which is bridged when the commands and interface mechanisms of an interactive system (in their case, VEs) match the intentions of a user. In the case of VEs, Norman's *interface mechanisms* can be specified as *VE User Interface Input Mechanisms* (e.g., glove, wand, 3D mouse). Norman also defines a *gulf of evaluation*, which is bridged when system output (presented via an interface display) provides an appropriate conceptual model that a user can readily perceive, evaluate, and understand. Norman's term *interface display* is mapped within the taxonomy to *VE User Interface Presentation Components*. They intentionally chose the term *presentation*, rather than *display*, to reflect the multimodal presentation

capabilities of VEs. These mappings are depicted in Figure 6 [Gabbard and Hix, 1998].

An important insight presented with the theory is the need to bridge the gulfs between *goals* and *physical system*. This notion is applicable within the taxonomy as well, emphasizing the bridging of *VE Users and User Tasks* and *The Virtual Model*. Thus, the four major areas shown in Figure 5 are strongly influenced by corresponding components of the theory of action, and the flow is strongly influenced by the theory's corresponding flow [Gabbard and Hix, 1998].

2. Accessing Supplemental VE Usability Resources via the Taxonomy

At the highest level, the taxonomy supports usability engineering as an analytical method to guide initial systematic reduction and refinement of the supplemental resources (e.g., guidelines, discussion, references). More specifically, taxonomy areas (graphically depicted in Figure 5) provide focused access to both usability guidelines and context-driven discussion [Gabbard and Hix, 1998].

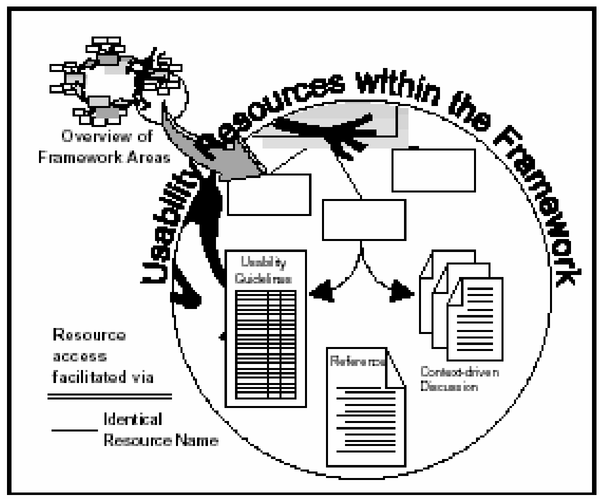


Figure 7. Accessing Specific Usability Design Guideline Tables and Context-Driven Discussion via the Taxonomy Usability Characteristics [From Gabbard and Hix, 1998].

In Figure 5, each of the four shaded boxes corresponds to both a collection of specific design guidelines (several tables) and the accompanying section of context-driven discussion. Each of the white boxes corresponds to a single table of this collection of specific guidelines and the corresponding context-driven discussion. Figure 7 graphically depicts how the taxonomy facilitates access to specific usability design guidelines and the corresponding context-driven discussion. In particular, access

to these resources is facilitated by identical resource naming. For example, both the table and context-driven prose associated with the taxonomy area *Object Manipulation* are labeled *Object Manipulation* [Gabbard and Hix, 1998].

In a hypertext document, selecting the taxonomy box labeled *Object Manipulation* would allow a reader to directly access either the specific usability guidelines associated with object manipulation, or the context-driven discussion on object manipulation.

a. Specific Usability Design Guidelines — Do's and Don'ts

Specific usability design guidelines — *do's and don'ts* for design and evaluation of VE user interfaces — are summarized in tables representing the first level of supplemental VE usability resource refinement. As previously mentioned, there is one table for each white box in Figure 5. They derived the guidelines from the *sources of inspiration*; these guidelines are further explained in lower levels of refinement, specifically context-driven discussion and its accompanying references (see next sub-sections b and c). There are currently 19 tables of specific usability design guidelines, all of which are available in the complete taxonomy document [Gabbard and Hix, 1998].

A portion of a usability design guideline table is shown in Table 1. This particular table addresses some general usability issues of VE user interface input mechanisms.

VE User Interface Input Mechanisms in General			
Label	**Usability Suggestion/Consideration**	**Page(s)[1]**	**Bibliography Ref(s)[1]**
Input1	Assess the extent to which degrees of freedom are integrabile and separable within the context of representative user tasks	98	[Jacob et al., 1994] [Zhai and Milgram, 1993b]
Input2	Eliminate extraneous degrees of freedom by implementing only those dimensions which users perceive as being related to given tasks	98	[Hinckley et al., 1994a]

[1] Note that *page numbers* and *references* given in example table do not refer to this document; rather *they* refer to the complete taxonomy document. *They* are included to illustrate table structure and content

Input3	Multiple (integral) degrees of freedom input is well-suited for coarse positioning tasks, but not for tasks which require precision	98	[Hinckley et al., 1994a]
Input4	When tasks require significant coordination and are not time critical (e.g., surgery), consider using deviation in three-space as a metric of device control (as opposed to time to target)	99	[Zhai and Sanders, 1997]
Input5	From the user's perspective, device output should be consistent with, and cognitively connected to, user actions	99	[MacKenzie, 1995]
Input6	For fine positioning tasks, employ low gain, for gross positioning tasks, high gain. When VEs contain both coarse and gross positioning tasks strive for a balance between the two determined by iterative user testing of representative positioning tasks	100	[MacKenzie, 1995]
Input7	Address possible effects that prolonged usage with particular input device(s) may have on user fatigue and task performance	100	[Zhai, 1995] [Card et al., 1991]
Input8	Decrease user cognitive load by avoiding devices such as joysticks and wands which, in effect, place themselves between users and environments	101	[Davies, 1996]
Input9	Input devices should make use of user physical constraints and affordances	101	[Norman and Draper, 1986] [Hinckley et al., 1994a]
Input10	Avoid integrating traditional input devices such as key-boards and mice in combination with 3D, free-space input devices (devices that move freely with users, as opposed to mounted or fixed devices)	101	[Hinckley et al., 1994a]

Table 1. Usability Design Guidelines Tables: VE User Interface Input Mechanisms [From Gabbard and Hix, 1997].

A table of guidelines also contains several different pointers to related sections in the context-driven discussion, pointed to by specific page numbers (*Page(s)***[1]**). *Bibliography Ref(s)***[1]** points to specific citations in the reference list. Thus, these tables (much like the taxonomy) serve as a resource map into additional detailed information found in the supplemental VE usability resources (namely, the discussion and references). *Label* in the tables is explained in sub-section b. Figure 8 depicts the connections available from

usability design guideline tables to relevant context-driven discussion and associated references [Gabbard and Hix, 1998].

It is important to realize that, although guidelines in each table are presented in an active tone, none of the guidelines should be taken or followed out of context. That is, the guidelines given in the tables are powerful, and most likely apply to particular arrangements of VE users, tasks, hardware, applications, etc. For example, one guideline reads,

> Eliminate extraneous degrees of freedom.

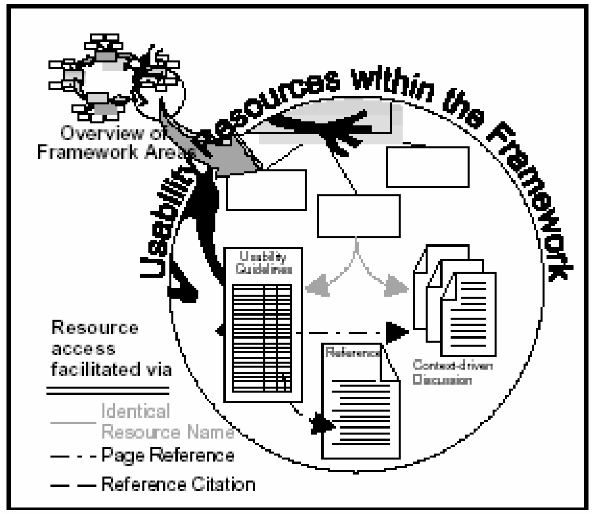


Figure 8. Accessing Context-Driven Discussion and References via Usability Design Guideline Tables [From Gabbard and Hix, 1998].

Clearly, to effectively use this guideline, a VE designer must know much more information about types of users, types of tasks, characteristics of the application, etc. Blindly applying the guidelines will not make a VE instantly usable. The purpose of subsequent refinement levels (i.e., context-driven discussion and reference list), discussed below, is to give the necessary context in which to assess and appropriately apply these usability design guidelines [Gabbard and Hix, 1998].

b. Context-Driven Discussion — Details of When and Why

The context-driven discussion provides readers with detailed information with which to assess appropriate application of usability guidelines. As dictated by the taxonomy's structure, context-driven discussion is presented in four sections — one for each major area of the taxonomy — each beginning with a general presentation of usability characteristics specific to that area. This is followed by an in-depth discussion of relevant usability-related issues and information to provide context for using specific usability design guidelines. At

this lower level of refinement, usability-related topics are addressed in terms of specific tasks, interaction techniques, hardware, etc. Issues are compared and contrasted, and — very importantly — apparent contradictions in research findings are elaborated. These discussions comprise the bulk of the complete taxonomy document (containing all supplemental VE usability resources), and are currently about 125 pages in length [Gabbard and Hix, 1998].

To facilitate non-linear access both into and out of the context-driven discussion, each mention of a VE usability characteristic is uniquely labeled (the *Label* in Table 1) and typeset in a special notation. For example, the textual discussion of the first guideline in Table 1 contains the label <<Input1>>, which is a pointer out of the context-driven discussion to this particular guideline in Table 1. Every label shown in the usability design guideline tables corresponds to a specific usability design guideline elaborated in the related context-driven discussion [Gabbard and Hix, 1998].

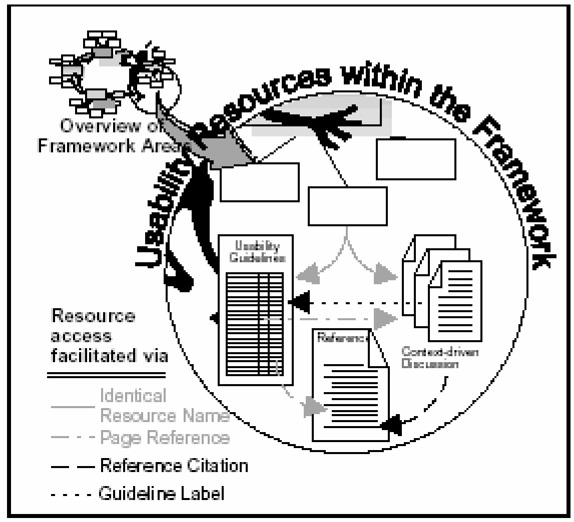


Figure 9. Accessing Specific Usability Design Guidelines and References via Context-Driven Discussion [From Gabbard and Hix, 1998].

Thus, guideline labels (in conjunction with page references) help readers find a particular segment of context-driven discussion when turning to the discussion. The guideline labels also help readers turn from the context-driven discussion back to the tables. Identical reference citations are found both in the tables and in the discussion. Access to and from the context-driven discussion is illustrated in Figure 9 [Gabbard and Hix, 1998].

c. Reference List — For More Information

Because the context-driven discussion contains, of necessity, only a few sentences about most references, a complete list of all cited references is included as a VE usability resource. Specifically, as mentioned, references are associated with particular VE usability design guidelines as well as the context-driven discussion. This list contains typical bibliographic information as well as WWW addresses when appropriate and available. It currently contains more than 100 citations, and is a rich resource in and of itself [Gabbard and Hix, 1998].

D. OBJECTIVES OF THIS RESEARCH

First of all, we want to emphasize and make clear that the taxonomy [Gabbard and Hix, 1997] is not our study. It is the Master Thesis of Joseph L. Gabbard done at the *Virginia Polytechnic Institute and State University* with Dr. Deborah Hix. This research — at least until now — is in paper/text form.

The taxonomy [Gabbard and Hix, 1997] can have both immediate and long-term impact on the field of VEs. In the short term, it comprehensively defines a structure of characteristics important for usability in VEs. The design space for VEs is far greater than that for traditional user interfaces such as GUIs. For example, VEs typically employ a suite of multimodal interaction devices with characteristics that are constantly emerging and changing. GUI devices, on the other hand, have matured into a steady state, exploiting the familiarity of the mouse and keyboard. Complexity and variation in VE interaction devices facilitate more complex and sometimes less predictable, user tasks [Gabbard and Hix, 1998].

At the highest level, the taxonomy supports usability engineering as an analytical method to guide initial systematic reduction and refinement of the supplemental resources (e.g., guidelines, discussion, references). More specifically, taxonomy areas (graphically depicted in Figure 5) provide focused access to both usability guidelines and context-driven discussion. In Figure 5, each of the four shaded boxes corresponds to both a collection of specific design

guidelines (several tables) and the accompanying section of context-driven discussion. Each of the white boxes corresponds to a single table of this collection of specific guidelines and the corresponding context-driven discussion. Figure 9 graphically depicts how the taxonomy facilitates access among specific usability design guidelines, the corresponding context-driven discussion and references. In particular, access to these resources is facilitated by identical resource naming.

The structure of the taxonomy is in non-linear form. It consists of usability characteristics, guidelines, context-driven discussion and references. Thus, when the end user needs to extract information from taxonomy to build or design their VEs, they may need to navigate the document from page to page. So the current paper/text form of the document has a navigation problem (see Figure 9). It is very annoying to go back and forth in the document.

In the long run, the taxonomy will, perhaps more importantly, provide a basic, scientific foundation for evolving a new generation of the methods for usability engineering of VEs. These new methods will come both from modification of existing methods so they accommodate VEs, as well as from altogether new approaches to usability engineering of VEs [Gabbard and Hix, 1998]. Thus, the taxonomy must be dynamic in order to add, delete and edit evolving new methods.

In order to overcome the navigation and dynamic property problem of the taxonomy, it seems reasonable to convert the taxonomy into dynamic hypermedia representation. When the taxonomy is converted into the dynamic web version, it is expected that the document will be more navigable, dynamic and readable. Therefore, to manage dynamic character of the taxonomy, Active Server Pages (ASP) will be used for extracting the data from a database. When updating the taxonomy, the database and related context-driven discussion pages will be updated.

Another important shortcoming of the taxonomy is that it is a snapshot of VE characteristics in time — 1997. It has covered the research results until 1997. On the other hand, VEs have not matured yet and still in the evolving phase. If we take another snapshot now and compare the results with taxonomy, we will find some inconsistencies: So the taxonomy must grow too. You must easily be able to change some parts, add new parts or remove parts when necessary. Hypermedia Representation of Taxonomy will support these features.

So in this study, the purpose is to build Hypermedia Representation of the Taxonomy and to evaluate the effectiveness of the user interface of it. The study will evaluate the entire interface, make recommendations to improve the interface and finally contain the redesigned interface. We will try to produce easy to learn and efficient user interface. User satisfaction is also one of our biggest goals.

E. SCOPE AND LIMITATIONS

The current taxonomy document will be transferred to a web application. The interface of this application will be improved by using iterative formative usability evaluation. After building the web site version of the taxonomy, it is expected that more people will access this source and use it. When using the web application they will save a lot of time. Lots of people will see it and make recommendations to refine it.

The taxonomy was built in 1997 and there have been lots of improvements in VE technology since that date. The content update of the taxonomy will be out of scope of this thesis.

II. LITERATURE REVIEW

A. OVERVIEW

The review of literature of this research includes journals, and textbooks covering the subjects of usability evaluation, human-computer interaction, and virtual environments. The purpose of this literature review is to provide an overview of the current theories and practices relating to usability evaluation on the methods used in this study to evaluate Hypermedia Representation of a Taxonomy of Usability Characteristics in VEs. As you will see later, in design phase, we used some guidelines that will be explained in Chapter III Section C — User Interface Design and they directed our design implementation. After design and implementation phase, the formative usability evaluation method from Hix and Hartson [1993] is used to evaluate the interface. We will widely try to explain this formative usability evaluation method in this chapter.

B. FORMATIVE EVALUATION

Formative Evaluation [Carroll and others, 1992; Dick and Carey, 1978; Scriven, 1967; Williges, 1984] is evaluation of the interaction design *as it is being developed,* early and continually throughout the interface development process. This is in comparison to *summative evaluation,* which is evaluation of the interaction design *after it is complete,* or nearly so. Summative evaluation is often used during field or beta testing, or to compare one product to another. For example, a summative evaluation of two systems, A and B, could show which one is better, where *better* is defined as *the user makes fewer errors with this one* or *the user subjectively prefers this one*. In practice, summative evaluation is rarely used for usability testing [Hix and Hartson, 1993].

On the other hand, formative evaluation, the mainstay of usability evaluation, is not to be confused with what is often thought of as typical human factors testing — for example, controlled hypothesis testing of an *m* by *n* factorial design with *y* independent variables, complete with quantitative data, statistical analyses, and numeric results. Controlled experimentation is valuable in

contributing to the science and principles of human factors but does not produce results in a time frame that meets the needs of the fast, cyclical iterative development process [Hix and Hartson, 1993].

In contrast, formative evaluation, performed in every cycle of iteration, produces quantitative data against which developers can compare the established usability specifications, and also produces qualitative data that can be used to help determine what changes to make to the interaction design to improve its usability. This *formative evaluation is begun as early in the development cycle as possible,* in order to discover usability problems while there is still plenty of time for modifications to be made to the design. By waiting until late in the development process, much of the interface will already be implemented, and it will be far more difficult to make changes indicated by usability evaluation [Hix and Hartson, 1993].

Summative evaluation is usually performed only once, near the end of the user interface development process. Formative evaluation is performed several times throughout the process; the rule of thumb is that an average of *three major cycles of formative evaluation, each followed by iterative redesign,* will be completed for each significant version of an interaction design. There may be additional very short cycles, to check out quickly a few small changes made to the interaction design, while the major cycles will be longer, to evaluate more extensive issues. You will typically get the most data from the first major cycle of evaluation. If the process is working properly and the user interaction design is indeed improving, later cycles will generate fewer new discoveries and will generally necessitate fewer changes in the design. The first cycle can generate an enormous amount of data, enough to be overwhelming. This chapter tells you how to collect and analyze these data in order to optimize the usability of the interface [Hix and Hartson, 1993].

Formative evaluation primarily addresses the path in the star life cycle between prototyping and design/ redesign. People sometimes mistakenly think that formative evaluation is not as rigorous or as formal as summative evaluation.

Actually, however, the distinction between formative and summative evaluation is not in its formality, but rather in the goal of each approach. Summative evaluation does not support the iterative refinement process represented in the star life cycle; waiting to evaluate an interface until it is almost complete will not allow much, if any, iterative refinement. Formative evaluation, because it is early and continual throughout the process, is most responsive to the iterative approach shown in the star life cycle (see Figure 10).

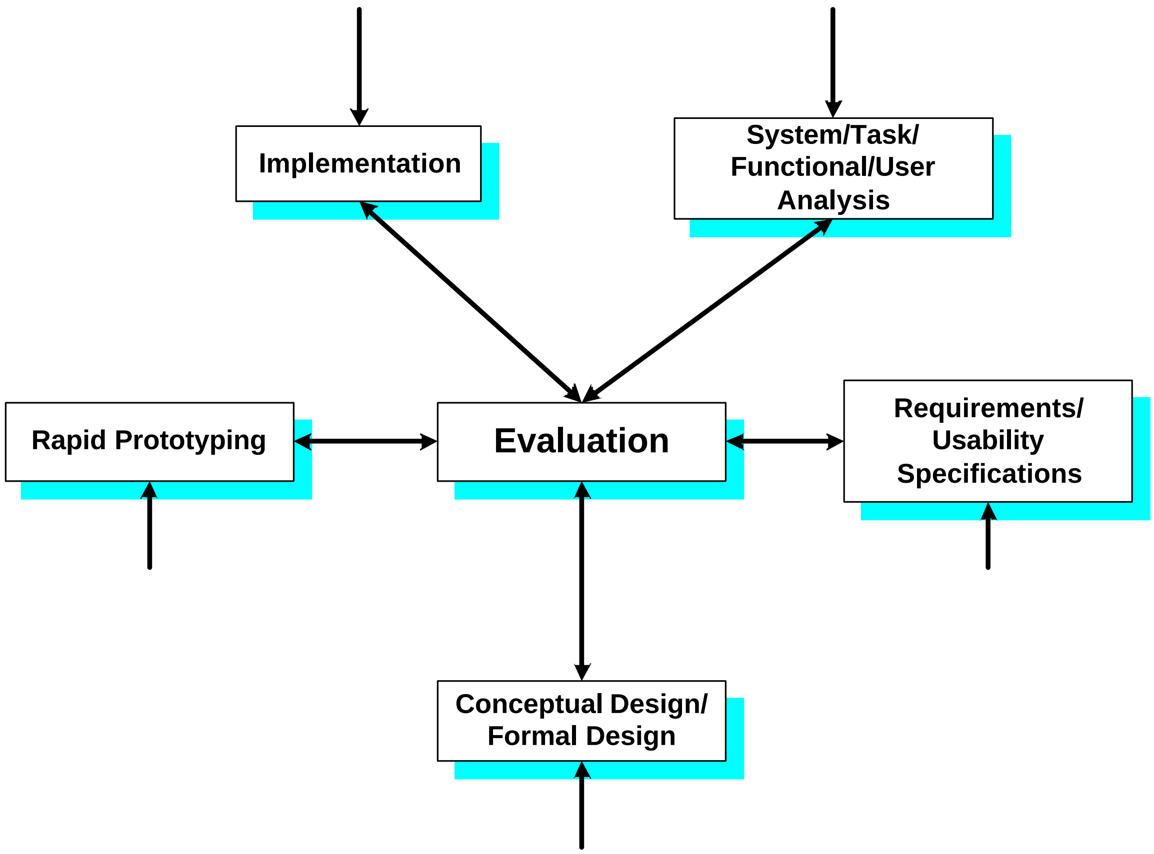


Figure 10. The Star Life Cycle Model [From Hix and Hartson, 1993].

It is important that members of the development team, and especially managers, understand this difference between formative and summative evaluation. Otherwise, because formative evaluation is not controlled testing and usually does not require many participants; your results may be discounted as being, for example, too informal, not scientifically rigorous, or not statistically significant. Formative evaluation is, indeed, rigorous and formal, in the sense of

having an explicit and well-defined procedure, and it does result in quantitative data but is not intended to address statistical significance. It does address the needs of users, and therefore of developers, to ensure high usability in an interface [Hix and Hartson, 1993].

Many people espouse a *10% rule* concerning evaluation: An interface development effort should have something that can be evaluated by the time the first 10% of the project resources (time and/or dollars) are expended. The previous chapter, on rapid prototyping, discussed how to quickly produce something testable; this chapter discusses in depth how to perform formative evaluation of early versions of the interaction design using prototypes [Hix and Hartson, 1993].

The bottom line is this: *Users will evaluate your interface sooner or later* — *either* correctly, in-house, using the proper techniques and under the appropriate conditions, *or* after it's in the field, when it is too late. Why not do it *right,* and evaluate it *sooner?*

1. Types of Formative Evaluation Data

Several types of data are generated during formative evaluation, each of which can be used in making decisions about iterative redesign of the user interface. The following types of formative evaluation data are discussed throughout the rest of this chapter [Hix and Hartson, 1993]:

- *Objective* — These are directly observed measures, typically of user performance while using the interface to perform benchmark tasks.
- *Subjective* — These represent opinions, usually of the user, concerning usability of the interface.
- *Quantitative* — These are numeric data and results, such as user performance metrics or opinion ratings. This kind of data is key in helping to monitor convergence toward usability specifications during all cycles of iterative development.

- *Qualitative* — These are nonnumeric data and results, such as lists of problems users had while using the interface, and they result in suggestions for modifications to improve the interaction design. This kind of data is useful in identifying which design features are associated with measured usability problems during all cycles of iterative development.

Even though people often associate objective evaluation only with quantitative data and subjective evaluation with qualitative data, subjective evaluation (e.g., using user preference scales or questionnaires) can also produce quantitative data. Also, objective evaluation activities (e.g., benchmark task performance measurements) can produce qualitative data (e.g., critical incidents and verbal protocol, discussed later in section E, on generating and collecting the data).

2. Steps in Formative Evaluation

The remainder of this chapter elaborates on details of the major steps in formative evaluation. These include the following [Hix and Hartson, 1993]:

- Developing the experiment
- Directing the evaluation sessions
- Collecting the data
- Analyzing the data
- Drawing conclusions to form a resolution for each design problem
- Redesigning and implementing the revised interface

While many members of the interface development team may be involved in performing these steps at various times, we refer to the person who is primarily responsible as the *user interaction design evaluator,* or just *evaluator,* for short.

C. DEVELOPING THE EXPERIMENT

Developing an experiment to be used for formative evaluation involves four main activities, not necessarily in the order given [Hix and Hartson, 1993]:

- Selecting participants to perform tasks
- Developing tasks for participants to perform
- Determining protocol and procedures for the evaluation sessions
- Pilot testing to shake down the experiment

1. Selecting Participants

One of your first activities related to formative evaluation is *evaluation participant selection* — determining appropriate users for the experimental sessions. *Participant* is the term that most recent human factors literature now uses to indicate a human taking part in an experiment. There are good reasons for this change in terminology; people, on hearing themselves referred to as *subjects*, will sometimes nervously joke about being attached to electrodes or ask to see the maze. It is better to *view the interface as the subject,* and the evaluation participant as helping you to evaluate the design.

The evaluator must determine the classes of representative users that will be used as participants to try out the interface. These participants should represent the typical kind of expected user of the interface being evaluated, including the users' general background, skill level, computer knowledge, application knowledge, and so on. Often, these attributes for expected user classes are explicitly stated in the usability specifications, and the participants should be chosen to match.

Appropriate users should be at least a little knowledgeable of the problem domain (e.g., word processing, accounting, graphical drawing, process control, airline reservations, or whatever the problem domain may be), but not necessarily knowledgeable of a specific interactive system within that domain. If an adequate user analysis was done up front (see [Hix and Hartson, 1993 — Chapter 5]), the evaluator will already have a good idea of the kinds of people who will fit the user profile to represent the various classes of users of the system being evaluated. If the user analysis was not sufficient, the evaluator can work

with marketing people and other members of the development team to help define more clearly the user profile and appropriate population.

The question arises, of course, as to where to find participants. Participants should not have to be coerced into taking part in an experiment, or they may come into it with a poor attitude and thereby color the results. Volunteers typically provide much better data. Often, people (coworkers, colleagues elsewhere in your organization, spouses, children, and so on) will volunteer their time to act as participants. Many organizations post notices in grocery stores or in other public places (e.g., libraries). Students at universities, community colleges, or even K-12, if appropriate, also work well. These people probably won't work for free; you will usually have to pay a modest hourly fee (for example, about a dollar above minimum wage is typical these days) in order to get the participants you need. In fact, it is always nice, and sometimes necessary, to offer payments/compensations to get participants. Various kinds of inexpensive compensations include mugs with your company logo, T- shirts of some sort, or even chocolate chip cookies! Use any and all of these strategies, as needed, to assemble the participant pool for evaluating your user interaction design. While it is often necessary to offer compensation in order to recruit participants, some practitioners believe that monetary rewards may bias results. For example, paid participants with greater financial need could be more motivated than participants without financial need [Hix and Hartson, 1993].

Another source you can use for finding participants is temporary employment agencies. A possible pitfall here: These agencies know nothing about usability evaluation, nor do they understand why it is so important to choose appropriate people as participants. These agencies' goal, after all, is to keep their pool of temporary workers employed. Particularly for potential participants sent from such an agency, as well as for those who respond to notices posted in public places, it is important to screen each person thoroughly to make sure each is appropriate for your current evaluation. You should have

developed a good user profile for anticipated users of your system by now; use this as the basis for screening potential participants [Hix and Hartson, 1993].

A common problem, particularly in a contractual development situation, is one in which an organization (e.g., a private company) is developing an interactive system under contract for a customer (e.g., some government agency). Sometimes, the customer — for whatever reasons — simply will not let the developer organization have access to representative users. The Navy, for example, can be rightfully hesitant about calling in its ships and shipboard personnel from the high seas to evaluate a system being developed to go on board [Hix and Hartson, 1993].

We do not have a magic solution to this problem but we can offer encouragement: If the organization producing the interface informs the customer, at the beginning of the interface development process, about how the process will proceed, it will then have the highest likelihood of getting representative users from the customer involved at appropriate times. In fact, rather in-depth discussions of the user interface development process are sometimes included in proposals in response to RFPs (requests for proposal) during the bidding process to award a contract. Customers are now beginning to look closely in the response to an RFP for an explanation of the process by which a potential bidder expects to develop a user interface. If these customers do not see terms such as *user analysis, formative evaluation, rapid prototyping,* and *iterative refinement* in the bid description, then the likelihood of that bidder getting the contract falls drastically. In fact, more and more customers are starting to demand a user interface development process of their contractors, as this process becomes more widely known and understood [Hix and Hartson, 1993].

When a customer knows up front exactly what to expect and approximately when to expect it, the customer is much more likely to cooperate and help provide appropriate participants for formative evaluation. However, it may still be difficult, in the beginning, to convince some customers that usability is crucial. Until the customer has personally observed a few evaluation sessions

or read the results of a formative evaluation cycle and seen changes made that improved usability, the customer may be unwilling to help much with providing participants [Hix and Hartson, 1993].

However, once the customer understands that the *success of the whole system revolves heavily around usability of the interface,* and that *usability of the interface revolves heavily around a development process involving usability testing,* the customer almost always will gladly supply the developer with appropriate participants. Once the customer sees the benefits of formative evaluation, the customer generally is very anxious to participate in any way possible to maximize its benefits. In addition, sometimes, when the customer has chosen a few representative users to be participants, these people have become so excited about the new system that lots of other people wanted to be participants, too — more people, in fact, than the formative evaluation schedule and resources could handle. The whole development process can, indeed, have a very positive effect on acceptance of a new interactive system by its customer [Hix and Hartson, 1993].

In addition to representative users, the *human-computer interaction expert* plays an important part in formative evaluation. Evaluators sometimes overlook the need for critical review of the interface by a human-computer interaction expert when developing a formative evaluation plan. An expert will be broadly knowledgeable in the area of interaction development and will have extensive experience in evaluating a wide variety of interfaces. In particular, this person should know a great deal about interaction design and critiquing, as well as all activities of the user interaction development process. This expert particularly needs to be familiar with interaction design guidelines [Hix and Hartson, 1993].

An expert does not necessarily have to know a great deal about the specific interactive system domain, but rather is interested in a more generic review of the interaction design. An expert will find subtle problems that a non-interface expert would be less likely to find (e.g., small inconsistencies, poor use of color, and confusing navigation). More importantly, a human-computer

interaction expert will offer alternative suggestions for fixing problems, unlike the representative user, who typically tends to find a problem but cannot offer suggestions for resolving it. An expert can draw on knowledge of guidelines, design and critiquing experience, and familiarity with a broad spectrum of interfaces, to offer one or more feasible, guideline-based suggestions for modifications to improve usability [Hix and Hartson, 1993].

What you do with a human-computer interaction expert during formative evaluation is somewhat different than what you do with participants representing typical users. Having the expert perform representative tasks, possibly your benchmark tasks, is a good place to start, but you probably do not want to time the expert or count the expert's errors. The expert is doing a critical review of the whole interaction design, so you typically will collect far more qualitative data than quantitative data during a review by an expert. If you give experts the benchmark tasks as a starting point, they may work through them all, or they may take their own path in exploring the rest of the interface. Either way will generally give you a great deal of valuable data to be used for design modifications — both problems in the design and guideline-based suggestions for improving the design. A word of caution: Do not think that a human-computer interaction expert can serve as a substitute for evaluation with representative users. You will get quite different data from the two different sources.

Nielsen [1992; Nielsen and Molich, 1990], in fact, espouses what he calls *heuristic evaluation* or *discount usability engineering,* which is related to the approach being described here. Heuristic evaluation is a technique for uncovering usability problems in a design by having a small set of participants (three to five) judge the compliance of the interaction design to a set of recognized usability guidelines (the heuristics). He has found, through empirical studies that human-computer interaction experts make the best participants, in terms of discovering usability problems, and such experts with knowledge of the problem domain of the interface being evaluated are even better than those who do not have this specific knowledge. Nielsen states that heuristic evaluation has

the advantages of being cheap, intuitive, and easy to motivate developers to do, and it is effective for use early in the development process.

You may be sitting there, saying to yourself,

> Right! These people are still crazy. There's just not time to do all this evaluation with bunches of participants.

Well, take heart. You don't need bunches of participants. You do need a few carefully chosen, really good representative users, and one or maybe two interaction experts. In fact, the purpose of formative evaluation is not to focus on a large number of experiments with a large number of participants for each one. Rather, it is to *focus on extracting as much information as possible from every participant who uses any part of the interface* [Carroll and Rosson, 1985; Whiteside and others, 1988].

As mentioned, some empirical work [Nielsen and Molich, 1990] has shown that the optimum number of participants for a cycle of formative evaluation is *three to five per user class.* Only one participant per class is typically not enough, but more than ten participants per class are not worth the diminishing returns obtained. After about five or six participants, they tend to cease finding new problems and mostly reiterate the ones already uncovered by prior participants. Often, three participants per well-defined user class is the most cost-effective number. The advice for getting started with usability specifications applies here again: *Start small.* Do a couple of cycles of testing with a couple of appropriate participants for your most representative user class. This is a perfectly manageable approach, and evaluators will become more skilled and more comfortable after going through the entire process a few times [Hix and Hartson, 1993].

A question that commonly arises is whether you should use the same participants for more than one cycle of formative evaluation. Suppose that you use three participants per cycle. The best approach to participant selection for successive evaluation cycles is typically to use, for each cycle after the first, one

participant from the previous cycle and two new participants. This way, you will get some feedback from the repeat participant on the reaction to the user interaction design changes from the previous cycle. You will also get a new set of data on the modified design from the two new participants [Hix and Hartson, 1993].

2. Developing Tasks

By now, the evaluator should have participated with other members of the development team in *identifying usability specification attributes and levels* (see [Hix and Hartson, 1993 — Chapter 8]). Because these specifications are the key to quantifiably — measurably — determining usability of the interface, they must be ready and waiting as a comparison point with actual results observed during formative evaluation sessions with participants [Hix and Hartson, 1993].

In addition to the *benchmark tasks* developed for the usability attributes, the evaluator may also identify other *representative tasks* for participants to perform. These tasks will not be tested quantitatively (that is, against usability specifications) but are deemed, for whatever reason, to be important in adding breadth to evaluation of the user interaction design. These additional tasks, especially in early cycles of evaluation, should be ones that users are expected to perform often, and therefore should be easy for a user to accomplish [Hix and Hartson, 1993].

In the early cycles of evaluation, these representative tasks might, for example, constitute a core set of tasks for the system being evaluated, without which a user cannot perform useful work. Just as with the benchmark tasks developed for testing usability attributes, additional representative tasks should, in general, be rather specific and should state *what* the user should do, rather than *how* the user should do it. Thus, if there is information about the design that is not related directly to usability specifications, but that an evaluator wishes to investigate, the evaluator can define any other desired tasks. The results of users performing those tasks will simply provide additional qualitative data for later analysis as input to the iterative refinement process [Hix and Hartson, 1993].

To prepare for an evaluation session, the evaluator should write down all tasks (both the benchmark and representative tasks) in the order in which a participant will be asked to perform them. However, the evaluator can administer the tasks to a participant in several different ways. The evaluator can either hand the participant the written list and ask the participant to work through each task before going on to the next one, or the evaluator can read each task out loud to the participant, one task at a time, waiting until the participant completes a task before going on to the next one. The evaluator can, of course, also use a combination of these two approaches; for example, giving the participant some tasks in writing and others orally. The nature of the tasks will help determine which approach is best, and pilot testing will help verify the choice. For example, if a task is fairly specific and contains detailed information (e.g., particular time, place, and person for an appointment), it is best to write out the tasks and hand them to the participant. If a task can be stated in only a few words that are easy to remember (e.g., Draw a rectangle; Go to the glossary; View Figure 3), then it may be appropriate to simply read each one aloud to the participant. In general, it is preferable let the participant read written tasks, ensuring that each participant is given exactly the same instructions. Asking a participant to read each task description aloud before beginning it helps the evaluator know when to start timing the task performance (i.e., when the participant has finished reading the task aloud) [Hix and Hartson, 1993].

In addition to strictly specified benchmark and representative tasks, the evaluator may also find it useful to observe the participant in informal *free use* of the interface, without the constraints of predefined tasks. In fact, this was included as a specific activity. To engage a participant in free use, the evaluator might simply say,

> Play around with the interface for awhile, doing anything you would like to, and talk aloud while you are working.

Free use is valuable for revealing participant and system behavior in situations not anticipated by designers, often situations that can break a poor

design. Ways in which to take verbal protocol, such as during free use, are discussed in section E, on generating and collecting the data [Hix and Hartson, 1993].

Benchmark tasks, other representative tasks, and free use are all key sources of critical incidents (on generating and collecting the data), a major form of the qualitative data to be collected. Free use by a participant can be performed after either some or all of the predefined tasks have been completed. Obviously, it should be performed after those tasks that are related to the initial use attribute [Hix and Hartson, 1993].

Training materials and documentation are other aspects of developing the tasks to be performed by participants during formative evaluation. If the evaluator anticipates that a user's manual or quick reference cards or any sort of training material will be available to users of the system, the use of these materials should be explicit in the task descriptions [Hix and Hartson, 1993].

Participants might be given time to read any training material at the beginning of the testing session, or they might be given the material and told they can refer to it, reading as necessary to find the desired information. The number of times participants refer to the training material, and the amount of assistance they are able to obtain from the material, for example, can also be important data about overall usability of the system [Hix and Hartson, 1993].

Documentation and training materials for a system should also be evaluated, of course. Realistically, however, most systems are complicated enough that it is too difficult to evaluate documentation and the interface in the same session. It is better to develop separate formative evaluation plans for the documentation, the training material, and the user interface; don't try to test more than one unknown at a time [Hix and Hartson, 1993].

3. Determining Protocol and Procedures

Finally, the evaluator must determine protocol and procedures for administering the experiment — exactly what will happen during an evaluation

session with a participant. The evaluator must decide on whether laboratory testing or field testing, or both, will be performed. *Laboratory testing* involves bringing the participant to the interface; that is, participants are brought into a usability lab setting where they perform the benchmark tasks, performance measures are taken as appropriate, free use is encouraged, and so on. *Field testing* involves bringing the interface to the participant; that is, the present version is set up in situ, in the normal working environment in which users are expected to use the interface, and more qualitative, longer-term data can be collected [Hix and Hartson, 1993].

Obviously lab and field testing each have pros and cons. In a laboratory setting, an evaluator can have greater control over the experiment, but the conditions are mostly artificial. On the other hand, in a field test, an evaluator has less control, yet the situation is more realistic. In general, laboratory testing yields more useful information for the earlier cycles of formative evaluation, when major problems with the interaction design are typically discovered. Field testing works well for later cycles when data on long-term performance with the interface desirable. A combination of the two is the ideal circumstance for formative evaluation, but in real life, true field testing may be limited or even impossible. In this case, laboratory testing may have to suffice [Hix and Hartson, 1993].

In conjunction with developing experimental procedures, the evaluator should prepare *introductory instructional remarks* that will be given uniformly to each participant. These remarks can be either written, to be read by the participant at the beginning of the experiment; or oral, to be read by the evaluator to the participant at the beginning of the experiment; or both. These remarks should briefly explain the purpose of the experiment, tell a little bit about the interface the participant will be using, state what the participant will be expected to do, and the procedure to be followed by the participant. For example, the instructions might state that a participant will be asked to perform some benchmark tasks that will be given by the evaluator, will be allowed to use the

system freely for awhile, then will be given some more benchmark tasks, and finally will be asked to complete an exit questionnaire [Hix and Hartson, 1993].

It is also important to specifically make clear to all participants that *the purpose of the session is to evaluate the system, not to evaluate them.* Some participants may be fearful that participation in this kind of test session will reflect poorly on them or even be used in their employment performance evaluations (if, for example, they work for the same organization that is developing the interface they are helping to evaluate), and they should be reassured that this is not the case. In this regard, it is effective to guarantee the confidentiality of individual information and anonymity of the data [Hix and Hartson, 1993].

The instructions may ask participants to talk aloud while working or may indicate that they can ask the evaluator questions at any time. The expected length of time for the evaluation session, if known (the evaluator should have some idea of how long a session will take after performing pilot testing), can also be included. The important point is that all participants be given uniform instructions at the beginning, and the easiest way to ensure uniformity is through written instructions. This way, all participants start with the same level of knowledge about the system and the tasks they are to perform. This uniform instruction for each participant will help ensure consistency and remove some of the potential variance from the test sessions [Hix and Hartson, 1993].

One final, but important, activity that should be emphasized here is the preparation of an *informed consent form* for each participant to sign. This form states that the participant is volunteering for the experiment, that the data may be used if the participant's name or identity is not associated with those data, that the participant understands that the experiment is in no way harmful, and that the participant may discontinue the experiment at any time. The consent form should also include any nondisclosure requirements. This is standard protocol for performing experiments using human participants, and protects both the evaluator and the participant. The informed consent form is legally and ethically required; it is not optional [Hix and Hartson, 1993].

There are experiments, of course, in which harm could come to a human participant, but the kind of experiments performed during formative evaluation of an interaction design are virtually never of this kind. (In fact, harm is more likely to come to the computer terminal, the evaluator, and/or the designers, inflicted by the participant frustrated by an interface with poor usability — fallout from a user melt-down!) The informed consent form is an obligation to the participant and a further indicator of the seriousness of the experiment. It is also a legal document to protect the organization performing the evaluation [Hix and Hartson, 1993].

4. Pilot Testing

Finally, once the benchmark tasks have been developed, the setting and procedures have been determined, and the types of participants chosen, the evaluator must perform some *pilot testing* to ensure that all parts of the experiment are ready [Hix and Hartson, 1993].

The evaluator must make sure that all necessary equipment is available, installed, and working properly, whether it be in the laboratory or in the field. Obviously, you do not want the hardware or software to crash during an experimental session. The experimental tasks should be completely run through at least once, using the intended hardware and software (i.e., the interface prototype) by someone other than the person(s) who developed the tasks, to make sure, for example, that the prototype supports all the necessary user actions and that the instructions are unambiguously worded [Hix and Hartson, 1993].

Because good representative participants may be hard to find, the evaluator will want to minimize the possibilities for problems that might invalidate a test session. It is very easy for an evaluator to inadvertently write a benchmark task in which the wording is unclear, and which can be misinterpreted by a participant during the experiment. For example, there is a subtle difference in the wording of the following two tasks: *Schedule an HCI meeting every Wednesday for one year, beginning on the next Wednesday* and *Schedule an HCI meeting every Wednesday for one year, beginning on next Wednesday*. In the first

wording, it is unclear whether a participant should schedule the weekly appointment beginning with whatever the next Wednesday from the *current* position on the calendar happens to be, regardless of today's date, or whether this implies, as the second wording intends, to schedule beginning on the next Wednesday from today. These kinds of problems can invalidate all data from a participant [Hix and Hartson, 1993]. (By the way, if you're still having trouble understanding these task descriptions after reading them several times, well, that's the point. Imagine how confused a participant might feel.)

Similarly, even more extensive pilot testing is needed prior to critical reviews by human-computer interaction experts. These experts do not work for free, and the evaluator will not want things going amiss during a session in which a hefty hourly fee is being paid for expert advice.

Sometimes, you will be pilot testing and evaluating a prototype that has known bugs and/or weaknesses. If this is the case, the best you can do is to include benchmark and representative tasks that avoid those problems as much as possible. However, nothing will ensure that a participant won't encounter them anyway, especially during free use. If the system does, in fact, blow up during an evaluation session, apologize to the participant, restart the system, and have the participant pick up where the crash occurred.

Test sessions will run much more smoothly and predictably if even a minimal amount of effort is put into pilot testing of procedures, hardware, software, instructions, and so on, in advance. Pilot testing requires a very small amount of time compared to all the other effort you put in setting up the experiment, and collecting and analyzing the data [Hix and Hartson, 1993].

D. DIRECTING THE EVALUATION SESSION

So far, you have all the details of your experiment worked out, including benchmark tasks, procedures, consent forms, and participant selection. It is finally time to bring a participant into the usability lab and get an evaluation session underway. The evaluator is responsible for making sure that the session runs smoothly and efficiently. Typically, the evaluator, during a formative

evaluation session, will be in the same room as the participant. For quantitative measures of performance, the evaluator should remain in the background, not interacting with the participant unless there is a problem. Sometimes even this is obtrusive, and the evaluator can be next door in a control room, if this is available. A video monitor and/or one-way mirror is helpful in this case for observing the session [Hix and Hartson, 1993].

For taking qualitative data, it is best to have the evaluator sitting beside the participant. This approach is sometimes termed *codiscovery,* in which an evaluator and a participant work together to uncover usability problems. In this situation, the evaluator must be cautious not to lead the participant so much that the evaluator interferes with the goals of the session or of collecting appropriate data [Hix and Hartson, 1993].

Usually, there is only one participant for a session, but occasionally, interesting data can be obtained from having two participants interact together while using an interface. Although the present discussion concentrates on how to direct the evaluation session with one participant, the same general procedures would apply to a session with two (or more) participants.

First, the evaluator should briefly show the participant the usability lab and equipment, including the other side of a one-way mirror, if there is one. The evaluator can also briefly explain the lab setup from the evaluator's viewpoint, if the participant is interested. The evaluator should next get the participant settled comfortably in front of the prototype, and then give the participant the written instructions related to the evaluation session. Once the participant has read and understood the instructions, the evaluator should get the participant's signature on the informed consent form. The evaluator should ask if the participant has any questions. When the participant is comfortable with the instructions, the evaluator can then commence with the evaluation portion of the session, according to the protocol and procedures worked out during experiment development and pilot testing [Hix and Hartson, 1993].

During the session, as the evaluator is administering the tasks and whatever else the participant is to do during the session, it may be necessary to prompt the participant, primarily during qualitative data collection, to obtain the desired information. For example, if the participant struggles for awhile with a particular task (on qualitative data generation techniques) without talking much, the evaluator might ask, *What are you trying to do?* or *What did you expect to happen when you clicked on the such-and-such icon?* or *What made you think that approach would work?* The evaluator may also ask such questions as *How would you like to perform that task?* or *What would make that icon easier to recognize?* [Hix and Hartson, 1993].

If, however, one of the objectives for formative evaluation is task completion and/or failure, the evaluator must be especially careful about the protocol for questioning and giving help to participants. The evaluator should, in general, *not* give a participant specific instructions on *how* to complete a task with which the participant may be struggling. By telling a participant the actions to perform, the evaluator obviously loses the information that would be acquired as a participant continues to attempt to accomplish the task [Hix and Hartson, 1993].

The first question an evaluator might ask could be something like, *Are you stuck?* or *Do you need a hint?* If the answer is *No*, the evaluator might then ask, *Please tell me what you are thinking* or *Please tell me what you are trying to do*. If the participant's answer is *Yes*, then a failure data point can be recorded and the evaluator can give help progressively. If the participant does ask for a hint, the evaluator might proceed, for example, by suggesting *Do you remember what you did before for such-and-such a task?* or *Do you see an icon (or a menu item or a button or whatever) anywhere on the screen that might help you perform the task?* or *Try using the help facility* — if there is one. The evaluator, however, should refrain from blatantly coaching the participant on *how* to perform a task [Hix and Hartson, 1993].

Even if a participant asks for specific help (What should I do now? or I'm really lost; can you help me?), the evaluator should, at most, give hints, such as

those just suggested, as to how to proceed. Sometimes, a participant will give up on a task, flatly stating *I quit*. When this happens, unless the evaluator can gently prod the participant into continuing to attempt the task, it is probably best to explain to the participant how to accomplish the task, lead the participant through the steps (especially if it is important to subsequent tasks that will be performed), and then let the participant go on to the next task. If participants become so disgusted that they want to quit the entire session, there is little an evaluator can or should do but thank them, pay them, and let them go [Hix and Hartson, 1993].

The evaluator should ask any question that is likely to extract a useful response from the participant, as long as the evaluator does not lead too much with the question. The evaluator, after all, will not have another chance to get information related to this session from this participant after the session is finished and therefore should maximize the qualitative data obtained by asking appropriate questions. With experience, evaluators become very creative at being appropriately evasive while still helping a participant out of a problem without adversely affecting the data collected. Evaluators also become more comfortable with phrasing and interjecting questions to the participant [Hix and Hartson, 1993].

Finally, when the participant has performed the desired tasks, including completion of any questionnaire (e.g., QUIS) or survey, the evaluator should answer any questions the participant may have, give the participant whatever reward has been determined (e.g., money, mug, T-shirt), then thank and dismiss the participant, concluding the evaluation session [Hix and Hartson, 1993].

E. GENERATING AND COLLECTING THE DATA

Once the evaluation session is underway, lots of interesting things quickly start happening between the participant and the interface being evaluated. The data you need to collect may start arriving in a flood. It can be overwhelming, but, by being prepared, you can make it easy and fun, especially if you know what kinds of data to collect. It is very easy for inexperienced evaluators to collect reams of data that are later virtually worthless as far as providing information

about improving the design and usability of the interface. To avoid this problem, let's look at the kinds of data that are most useful in helping us measure and achieve our usability goals. There are methods for generating and collecting both qualitative and quantitative data, discussed in the following sections [Hix and Hartson, 1993].

1. Quantitative Data Generation Techniques

Quantitative techniques are used to measure directly the observed usability levels, in order to compare them against the specified levels set in the usability specifications. There are two main kinds of quantitative data generation techniques most often used in formative evaluation:

- Benchmark tasks
- User preference questionnaires

The development of *benchmark tasks* has been discussed extensively in [Hix and Hartson, 1993 — Chapter 8]. During the experiment, each participant performs the prescribed benchmark tasks, and if appropriate, the evaluator takes numeric data, depending on what is being measured. For example, the evaluator may measure the time it takes the participant to perform a task, or count the number of errors a participant makes while performing a task, or count the number of tasks a participant can perform within a given time period. Again, remember the need for pretesting the benchmark tasks, to make sure that they are clearly stated for the participants, and also to make sure that the metrics they are intended to produce are practically measurable. Counting the number of tasks in either five seconds or five hours, for example, is not reasonable.

Counting errors sounds, on the surface, as if it would be straightforward. However, it can be rather tricky. The main difficulties are in deciding what constitutes an error, and also in recognizing that an error is occurring in real time during an evaluation session. There are several effective approaches for recognizing errors. In general, an error is a special case of a critical incident (see sub-section 2, on qualitative data generation techniques). Any time a participant

cannot take a task to completion, an error (at least one, probably more) has occurred [Hix and Hartson, 1993].

Another kind of error can be identified when the participant does something wrong — namely, taking *any action that does not lead to progress in performing the desired task.* Note that this definition does *not* count accessing online help or other documentation as an error. Another way to think of this would be that a participant takes a *wrong turn along the expected path of task performance,* such as choosing the incorrect item from a menu or selecting the wrong button, and these choices do not lead to progress in performing the desired task [Hix and Hartson, 1993].

Sometimes, a participant takes a wrong turn and then *later backs up;* sometimes successfully (i.e., still is able to take the intended task to completion) and other times not successfully. In either case, an error (or errors) has still occurred. However, it is important to note the circumstances under which the participant attempted to back up, and whether the participant was successful in figuring out what was wrong. There are also incidents when a participant does something you did not expect, something that might initially appear to be a wrong turn but ends up being a different way to accomplish a task than you had in mind. This does not generally constitute an error but still could be considered a critical incident [Hix and Hartson, 1993].

Error making and error recovery during a session are also a chance for the evaluator to take data on how much time a user spends dealing with errors. These data are used later in impact analysis (in sub-section 3, on the effects on user performance). Often, however, it is difficult to know exactly when an error situation has begun. Some are quite obvious, while you may not recognize others as errors until the participant has progressed further along a fruitless path and is therefore well into an error situation. Thus, it can be difficult to capture, in real time, the time spent in making and dealing with errors. You may not recognize that an error is occurring in time to start a timer. A note of the current video-frame

counter (if available to you) at this point will facilitate obtaining these data by selective review of the videotape after the session [Hix and Hartson, 1993].

The second quantitative data generation technique is *user preference questionnaires,* or semantic differential scales. These fancy terms refer to something you are already familiar with — namely, categorical rankings (e.g., from 0 to 9, or -2 to 2, or never to always, or strongly agree to strongly disagree) for different features that, in this case, are relevant to the usability of the interface being evaluated. This kind of questionnaire or survey is inexpensive to administer but not easy to produce so that the data are valid and reliable. Questionnaires are the most effective technique for producing quantitative data on subjective user opinion of an interface. The QUIS survey (see [Hix and Hartson, 1993 — Chapter 8]) is one of the most comprehensive and readily available of these validated questionnaires [Hix and Hartson, 1993].

Even these simple measuring instruments are, however, not without problems. For example, the phenomenon termed the *halo effect* sometimes occurs with user preference questionnaires: Participants will give unreasonably good rankings to an interface. This happens for a variety of reasons: Some people want to be nice; others don't want to be negative; some are looking for jobs. However, there is also the *pitchfork effect,* in which participants give unrealistically low rankings. Perhaps they're having a bad day, or they had a fight with their spouse, or they don't feel appreciated in their job and want to cause trouble. There is really very little way to control for these two phenomena across your participants. You can discard data from any participant you think is not cooperating or otherwise properly participating in the evaluation. The most important suggestion is to be aware of the possibility and to be consistent in collecting and analyzing the data from user preference questionnaires [Hix and Hartson, 1993].

2. Qualitative Data Generation Techniques

Qualitative data are sometimes more mysterious and elusive than quantitative data. However, qualitative data are extremely important in

performing formative evaluation of a user interaction design for usability. The kinds of techniques that are most effective for generating qualitative data include the following [Hix and Hartson, 1993]:

- Concurrent verbal protocol taking
- Retrospective verbal protocol taking
- Critical incident taking
- Structured interviews

Perhaps the most common technique for qualitative data generation is verbal protocol taking, sometimes also called *thinking aloud*. This approach is immensely effective in determining what problems participants are having and what might be done to fix those problems. In *concurrent verbal protocol taking,* the evaluator asks participants to talk out loud while working during an evaluation session, indicating what they are trying to do, or why they are having a problem, what they expected to happen that didn't, what they wished had happened, and so on [Hix and Hartson, 1993].

This technique obviously is invasive to a participant, so unless the participant, offers it naturally, the evaluator should not actively elicit it for benchmark tasks where timing data are being taken. However, there is evidence that, except for very low-level tasks that occur in a very short time (a few seconds), thinking aloud does not measurably affect task performance. This is especially true if the participant is just thinking aloud and not being interrupted much by questions from the evaluator. So the verbal protocol technique is frequently employed during free use of the system, but it can also be effective during performance of timed tasks [Hix and Hartson, 1993].

The evaluator will find that some participants are not good at thinking aloud while they work; they will not talk much, and the evaluator will have to prod them constantly to find out what they are thinking or trying to do. For tasks that are not timed, it is perfectly acceptable for the evaluator to query such reticent

talkers, in order to discover the desired information. The previous section (section D, on directing the evaluation session) discussed various ways to prompt a reticent talker. Remember, one of the goals in formative evaluation is not to have a large number of participants, but rather to extract as much data as possible from each and every participant. Evaluators become more skilled at this as they work with more participants [Hix and Hartson, 1993].

For *retrospective,* or *post hoc, verbal protocol taking,* the evaluator lets participants work relatively uninterrupted during a taped session, rather than prodding them to think aloud very much. Then, immediately after the session, the evaluator and each participant review the videotape together, and the evaluator asks the participant to analyze what was occurring during the session. The assumption here is that a participant is at least as good as an evaluator in analyzing the data, especially if guided with appropriate questions by the evaluator during the videotape review. This postsession discussion and questioning does not interfere in any way with real-time task performance or collection of timing data. Analyzing verbal protocol data that are collected by an evaluator during an evaluation session can force the evaluator to make assumptions, guesses, and interpretations about what the participant was really thinking or trying to do. In retrospective verbal protocol taking, an evaluator can find out directly from participants what they were thinking, without having to guess or infer it [Hix and Hartson, 1993].

Retrospective verbal protocol taking works well with participants who have trouble performing tasks while simultaneously verbalizing what they are trying to do and/or what they are thinking. However, its biggest drawback is time and procedural constraints. It generally takes a minimum of three hours with a participant to conduct an evaluation session and then to follow it with retrospective analysis of a videotape. Also, it can take much longer than this, depending on the length of the actual evaluation session and the level of analysis given by the participant [Hix and Hartson, 1993].

You don't have to look at everything on the tape during the postsession review. Nonetheless, there are usually a large enough number of interesting incidents that you need to analyze with the participant that it typically takes at least twice as long to perform the retrospective analysis as it took for the session itself. It is very important to hold the review immediately after the session because the insights and ideas of the participant about the interface are very ephemeral and will be forgotten quickly. Retrospective verbal protocol taking is a good example of the codiscovery approach mentioned in section D — directing the evaluation session [Hix and Hartson, 1993].

During verbal protocol taking, you will find that many participants are able to express clearly what they don't like about an interaction design, but they often do not know what suggestions to make for changes. Some participants will, however, come up with a suggestion as an alternative design to something they don't like that will make the development team wonder why they didn't think of it earlier. Don't count on this happening very often, but this phenomenon can occur with both concurrent and retrospective verbal protocol taking [Hix and Hartson, 1993].

Despite its popularity and usefulness, verbal protocol is not without its controversies. In particular, it is an invasive data generation technique, and if not properly handled by an evaluator, it can affect the data collected. It is easy to get people to rationalize anything they experience, and they can be easily convinced, especially by an unskilled evaluator, that the problems they had with the design were not so bad, after all, or that they just misunderstood the design or the task description or whatever [Hix and Hartson, 1993].

Verbal protocol helps uncover the working knowledge and assumptions of a typical user, which help not only to uncover a usability problem but also to provide reasons as to why a specific incident occurred. It helps determine what information or knowledge a user was missing that would have allowed the user to successfully complete a task.

Another kind of qualitative data generation that is important, often in conjunction with verbal protocol taking, is *critical incident taking* [del Galdo and others, 1986]. A *critical incident* is something that happens while a participant is working that has a significant effect, either positive or negative, on task performance or user satisfaction, and thus on usability of the interface. Critical incident data help focus analysis of the qualitative data, especially the verbal protocol data.

A bad, or *negative,* critical incident is typically a problem a participant encounters — something that causes an error, something that blocks (even temporarily) progress in task performance, something that results in a pejorative remark by the participant, and so on. For example, an evaluator might observe a participant try unsuccessfully five times to enlarge a graphical image on the screen, using a graphics editor. If it is taking the participant so many tries to perform the task, it is probably an indication that this particular part of the design should be improved. Similarly, the participant may begin to show signs of frustration, either with remarks (e.g., What is this thing doing?, Why did it do that?, Why won't it do what I tell it to?) or actions (e.g., shaking a fist at the screen, shrugging shoulders defeatedly, drumming fingers impatiently on the table, or uttering various four-letter words) [Hix and Hartson, 1993].

An occurrence that causes a participant to express satisfaction or closure in some way (e.g., That was neat!, Oh, now I see., Cool!) is a good, or *positive,* critical incident. When a first-time participant immediately understands, for example, the metaphor of how to manipulate a graphical object, that can also be a positive critical incident. While negative critical incidents indicate problems in the interaction design, positive critical incidents indicate metaphors and details that, because they work well or a participant likes them, should be considered for use in other appropriate places throughout an interface. Critical incidents can be observed during performance of benchmark tasks, other representative tasks, or when a participant is freely using the system [Hix and Hartson, 1993].

Structured interviews [Hix and Hartson, 1993] provide another form of qualitative data. These are typically in the form of a postexperiment interview, a series of preplanned questions that the evaluator asks each participant. A typical postsession interview might include, for example, such general questions as *What did you like best about the interface?*, *What did you like least?* and *How would you change so-and-so?*. An interesting question to ask is *What are the three most important pieces of information that a user must know to begin using this interface?* For example, in one design, some of the results of a database query were presented to the user as small circles. Most users did not at first realize that they could get more information if they clicked on a circle. So one very important piece of information users needed to know about the design was that they should treat a circle as an icon, and that they could manipulate it accordingly.

The interview questions may be asked by the evaluator, who writes down (or otherwise records) a participant's answers, or a participant may fill out the interview questionnaire. There is a danger of constructing an interview that will not produce valid and reliable data; it is therefore necessary to produce such a set of interview questions with assistance from someone who is skilled in interview development [Hix and Hartson, 1993].

3. Data Collection Techniques

So far, this chapter has described ways of generating various kinds of data to collect, but not *how* to collect them. There are several recommended techniques for capturing both qualitative and quantitative data from participants during a formative evaluation experiment, including [Hix and Hartson, 1993]:

- Real-time note-taking
- Videotaping
- Audiotaping
- Internal instrumentation of the interface

With various experience and numerous conversations with other evaluators indicate that *real-time note-taking* is still the most effective technique to use for data (especially qualitative) capture during a formative evaluation session. The evaluator should be prepared to take copious notes as activities proceed during a session. When an evaluator is directing a test session for the first few times, it is a good idea to have a second evaluator also observing the session in order to help take notes. The primary evaluator is responsible for giving instructions, prompting the participant, administering appropriate tasks, timing tasks when necessary, and taking notes on the entire procedure. Until an evaluator is comfortable with this multitude of simultaneous activities that can happen quickly during an evaluation session, another person with the specific responsibility of taking notes and perhaps timing task performance can be invaluable. Even after becoming experienced with all aspects of directing an experiment, an evaluator may still find it helpful to have another evaluator observing the session, especially if the session is expected to be rather lengthy, say an hour or more [Hix and Hartson, 1993].

To capture observations and notes, an evaluator can use either *pencil and paper* or *computer tools* such as word processors and/or spreadsheets. Many evaluators find that they can type data into a computer much faster than they can write (legibly). Then, during data analysis, even using a word processor's search facilities for such time-consuming activities as locating and counting similar incidents can be a huge time-saver. Using the computer may be more awkward than paper-and-pencil note-taking when the evaluator is in the same room as the participant. However, if the evaluator is using a laptop, or notebook computer, it seems to be much less invasive to the participant than a full-sized personal computer or workstation. The evaluator can explain why the computer is being used as part of the lab tour at the beginning of the evaluation session. Additionally, a person in a control room next door with a video monitor or one-way mirror could use a computer to take notes unobtrusively [Hix and Hartson, 1993].

To collect quantitative data, the required equipment is minimal. Each evaluator who will be timing task performance by the participants will need a stopwatch or a clock with a seconds hand, and some kind of tally sheet for noting and/or counting errors, timings, and other observations. The simplest approach to capturing these data is to use a form such as shown in Figure 11, which has a column specifically for noting errors associated with each task. These forms can be either reproduced on paper or set up in advance in a word processor or a spreadsheet [Hix and Hartson, 1993].

PARTICIPANT ID:		**Session Date:** **Session Start Time:** **Session End Time:**			
Task Description	**Tape Counter**	**No. of errors**	**Elapsed Time**	**Participant's Actions and Comments**	**Evaluator's Observations**
A Schedule...					
B					

Figure 11. Sample Form for Collecting both Quantitative and Qualitative Data during an Evaluation Session [From Hix and Hartson, 1993].

To collect qualitative data, the evaluator (or evaluators) should note all observed critical incidents, as well as any other observations, as a participant performs each task or uses the interface freely. A simple form such as shown in Figure 11 is useful to help structure the data collection. The evaluator should fill in the predefined tasks in the Task Description column before an evaluation session begins, leaving quite a bit of space between each one. The evaluator can also fill in the participant ID and session date before a session begins. This form can be used to record errors in the No. of Errors column, and elapsed time for task performance in the Elapsed Time column (when these are relevant measures for the task being performed). These values can then be later related to usability specifications as appropriate [Hix and Hartson, 1993].

If the videotaping setup has a frame counter or timing device, the evaluator can use the Tape Counter column to note the frame number or time associated with a particular task, action, comment, or observation. The

Participant's Actions and Comments column will contain many of the critical incidents for each task. Often, direct quotes from participants are effective and easy to capture. (These also make good video clips for selling these ideas outside the lab.) The Evaluator's Observations column can be used to record any other interesting information (e.g., an idea for a design fix for an observed problem). Comments and observations may be lengthy, especially for complicated tasks. They describe the critical incidents that will be used to detect both problems and good features during the data analysis step of formative evaluation (see analyzing the data section). You can also use this same form during free use [Hix and Hartson, 1993].

Videotaping [Hix and Hartson, 1993] is a well-known and frequently used data collection technique. Many usability labs have an elaborate multicamera videotaping setup, with split-screen monitor for recording/editing capability, frame-accurate time tracking, and so. Videotaping has many advantages, including the capture of every detail that occurs during an evaluation session. If multiple cameras are used, one can be aimed, for example, at the participant's hands and the screen, another at the participant's face, and perhaps a third can be capturing a wide-angle view of evaluator, participant, and computer. Generally, one camera is adequate, and more than two cameras may be excessive. A camera aimed at the participant's hands and the screen is the most important, and a second, if available, should be aimed for a broader view, including the participant's face.

Some people often ask,

> Well, why not capture as much on tape as possible; you don't have to analyze it all if you don't want to.

This is true, but the problem with analysis of videotape is twofold. First, it can take as much as eight hours to analyze each one hour of videotape [Mackay and Davenport, 1989]. The chances of someone laboriously going back through several hours of videotape from half a dozen evaluation sessions is therefore very slim. Second, with multiple views and/or tapes of the same test session,

there is a problem of synchronization of the tapes (e.g., was the participant grimacing when she was trying to move the icon, or when it disappeared unexpectedly just after she tried to move it?) [Hix and Hartson, 1993].

There is really no point in using two (or more) cameras unless you have very sophisticated (read: expensive) equipment to merge two views onto one tape, alleviating the second problem. Even so, the first problem remains. The *main use of videotape should be as a backup* for what happened during an evaluation session, not as the main source of data to be captured and analyzed [Hix and Hartson, 1993].

The Tape Counter column shown in Figure 11 is invaluable when the video-tape is used as a backup. Sometimes, during an evaluation session, things happen so fast that, even with two evaluators taking notes, it simply isn't possible to write down everything of interest that is going on. When this happens, the Tape Counter column provides a pointer back into the videotape. The evaluators can, after the session, go back to each such place on the videotape and review it efficiently at their leisure, and without the real-time stress of continuing the session in an orderly fashion. For example, in case of confusion, uncertainty about a specific detail, or some missed part of a critical incident that occurred during an evaluation session, the evaluator can — if the tape counter value was noted — quickly go to a specific point on the videotape and review a very short sequence to collect the missing data. If the tape counter value was not noted, then the evaluator can, of course, search for the desired spot on the tape, but this can obviously take much more time. There are some tools to make reviewing videotapes more efficient, and, when used, the usefulness of the videotape goes way up, but so does the cost of the equipment [Hix and Hartson, 1993].

A few *carefully selected video clips (say, of five minutes each or less) can be of great influence on a development team that is resistant to making changes* to what the team members believe to be their already perfect design. Sometimes, programmers who have the major responsibility for an interaction design watch video clips in awe while a bewildered participant struggles to perform a task with

an awkward interface. Interestingly, their response is sometimes *What a stupid user!* rather than the appropriate *Wow, do we need to work on that interaction design!* Fortunately, as an awareness of the importance of usability increases, such inappropriate comments are heard less and less. These same video clips can also be useful in convincing management that there is a usability problem in the first place [Hix and Hartson, 1993].

Audiotaping [Hix and Hartson, 1993] of test sessions should be done when videotaping is not available (e.g., in field testing). It, too, should be used only as a backup, and not as the main data capture technique with the expectation of later going back and analyzing the full audiotaped session. While it does not capture the visual aspects of the test session, the oral exchanges that take place between an evaluator and a participant can be very valuable for later data analysis.

You probably are wondering just how much may be missed by an evaluator trying to take all the notes for an evaluation session in real time, without going back to review the videotape. Hix and Hartson [1993] wondered this, too, and performed some simple studies to try to determine how much could be captured by evaluators taking real-time notes versus a complete review of the videotape. In one study, for example, two experienced evaluators observed an evaluation session of about two hours, capturing comments and observations by writing them down. The entire session was also videotaped, and a third experienced evaluator reviewed the videotape to capture comments and observations. The third evaluator could go back and forth and review any portion of the videotape as many times as desired. It took the third evaluator more than 12 tedious hours, over a 2-week period, to analyze the videotape in detail. The results were then compared from the real-time data collection to the data collected in the videotape review. On average, the postsession detailed videotape analysis resulted in an increase of observed critical incidents of no more than 10% over the real-time critical incident capture. Also, almost without exception, these few incidents were minor ones that had no real impact on the

usability of the interface. They concluded, therefore, that postsession detailed videotape review has drastically diminishing returns for the amount of increased, useful data it provides. Thus, it appears that *real-time note-taking (either with pencil and paper or computer) is the most efficient means* of data capture during usability evaluation sessions [Hix and Hartson, 1993].

Finally, another useful way to capture the kinds of data discussed in this chapter is *internally instrumenting the interface* being evaluated to capture individual events, from user keystrokes and mouse clicks to start and stop times of routines associated with specific tasks. For example, data on user errors or frequency of command usage, or elapsed task times taken from start-stop times, can be automatically collected by a fairly simple program. There is, however, a potential problem with this technique: what to do with the collected data. Evaluators, especially novice ones, may think *the more data, the better*, but then find themselves inundated with details of keystrokes and mouse clicks. A fairly short session, say half an hour, can produce a several-megabyte user session transcript file. Manual analysis of a file dump printed as a 10-inch high (or even 10-foot high) stack of paper is totally untenable [Hix and Hartson, 1993].

The difficult question is, *What analysis should be done once such data are extracted from a transcript file? How can, for example, any of these keystrokes or cursor movements be associated with anything significant, good or bad, happening to the participant, and therefore related to usability? What do they mean in terms of the usability of the interface? What do they imply for the next iteration of modifications?* [Hix and Hartson, 1993].

The only feasible way in which such data might be useful is if their analysis can be automated, and there appear to be very few workable techniques for analyzing (either manually or automatedly) user session transcripts. One such technique is Maximal Repeating Patterns, or MRPs [Siochi and Ehrich, 1991], in which repeating user action patterns of maximum length are extracted from a user session transcript, based on the hypothesis that repeated patterns of usage (e.g., sequences of repeated commands) contain *interesting* information about

an interface's usability. In fact, this technique was compared empirically to observational evaluation of an interface [Siochi and Hix, 1991]. Most problems discovered by observing participants of an interface were found independently by MRP analysis of user session transcripts. However, the MRP technique, too, produces voluminous data, and only a prototype tool for automated evaluation exists. Also, while the MRP technique does help to pinpoint specific problems, it does not indicate how the interaction design should be modified to fix those problems [Hix and Hartson, 1993].

There are a few advantages of collecting user action data via instrumenting an interface. It can be employed in situ, thereby collecting real user data in field evaluation, which typically better represents a user's actual work context than data collected during laboratory evaluation. Collection of data via instrumentation is noninvasive (assuming it does not perceptibly slow down the system). This kind of data collection is cheaper than observational data because data can be automatically collected at multiple field sites without the need for dispatching platoons of evaluators to each site. However, until the information relating to usability that such data provide is better understood, and until satisfactory tools for automating such analysis are developed, its use is far less effective than direct observation of representative users, both in lab and field sites, for collecting data that will most influence the usability of an interface. We do not believe that any kind of analysis of user session transcripts will ever completely replace the kind of formative evaluation, involving observations of representative users, as described here [Hix and Hartson, 1993].

F. ANALYZING THE DATA

After all evaluation sessions for a particular cycle of formative evaluation are completed, the data collected during those sessions must then be analyzed. In general, evaluators do not perform inferential statistical analyses, such as analyses of variance (ANOVAs) or t-tests or F-tests. Rather, they use data analysis techniques that will help determine whether the interface has met the usability specification levels, and if it has not, analysis indicates how to modify

the design to help in converging toward those goals in subsequent cycles of formative evaluation [Hix and Hartson, 1993].

At this point in the iterative cycle comes a major decision: Accept the interaction design as it is, or consider a redesign. This decision must be made at a global —interface metaphor— level, as well as a detailed —individual problem— level. To help make this decision, the data collected must be analyzed.

The first step in analyzing the data is to compute averages and any other values stated in the usability specifications for timing, error counts, questionnaire ratings, and so on. A word of caution: Computing only the mean to determine whether usability specifications have been met can be misleading, because the mean is not resistant to outliers. With a small number of participants such as are typical in formative evaluation, it is possible for a mean to meet a reasonable preestablished usability specification, while there are serious usability problems. In fact, outliers may indicate serious usability problems. To help compensate for this, you may want also to report the standard deviation, and maybe the median [Hix and Hartson, 1993].

Next, enter a summary of your results into the usability specification table and decide your next step. If all worst acceptable levels have been met and enough planned target levels been met to satisfy the development team that usability of the present version of the interaction design is acceptable, then the design is satisfactory, and you can stop iterating for this version [Hix and Hartson, 1993].

The one exception to terminating iteration when the minimum levels have been is if, for whatever reason, you suspect that your usability specifications may be too lenient and therefore not a good indicator of high usability. For example, in a situation where all planned target levels were met or exceeded, but observations during evaluation sessions showed that participants were frustrated and performed tasks poorly, your intuition will probably tell you that the interface

is, in fact, not acceptable in terms of its usability, despite having met all the specified goals. Then, obviously, the development team should reassess the usability specifications to see whether they should be more (or less) stringent [Hix and Hartson, 1993].

In most cases where all usability specifications are met, though, you can stop iterating; you have reached the desired level of usability for the present version of the system. if you have not met your usability specifications (the most likely situation after the first cycle of testing), then you should continue with more in-depth data analysis, as described later.

The goal in further data analysis —much of which is qualitative data analysis— is structured identification of the observed problems and potential solutions to them. The subsequent activities address solving those problems in order of their potential impact on usability of the interface. The process of determining how to convert the collected data into scheduled design and implementation solutions is essentially one of negotiation in which, at various times, all members of the development team are involved [Hix and Hartson, 1993].

In order to make final decisions, developers must also know the total amount of time allocated to making design changes for the current cycle of iteration. To do those developers should look for impact, and/or cost/importance analysis (see [Hix and Hartson, 1993] for more info about impact, cost and importance analysis).

G. DRAWING CONCLUSIONS TO FORM A RESOLUTION FOR EACH PROBLEM

Finally, after impact analysis and/or cost/importance analysis of all problems in the list, developers must make a *resolution —a* final decision— about each problem. This is an indication of *how each problem will be addressed* (e.g., do it; do it, time permitting; postpone it indefinitely) and which solutions will be implemented [Hix and Hartson, 1993].

Problem	Effect on User Performance	Importance	Solution(s)	Cost	Resolution
Too much window resolution	10 to 35 minutes	High	Fix window placement automatically, but allow user to reposition it	6 hours	
Black arrow on black background	N/A	Low	Reverse arrow to white on black	1 hour	

Table 2. Data from Formative Evaluation of a Graphical Drawing Application [From Hix and Hartson, 1993].

Having done both some impact and cost/importance analyses, at last, the Resolution column of Table 2 can be completed. In fact, from the list ordered by importance (high to low) and, within that, cost (low to high), with high importance/low cost at the top of the list followed by high importance and moderate/high cost, you can determine the optimum choice of problems to address, given the time and other resources allotted for modifications (see Figure 12) [Hix and Hartson, 1993].

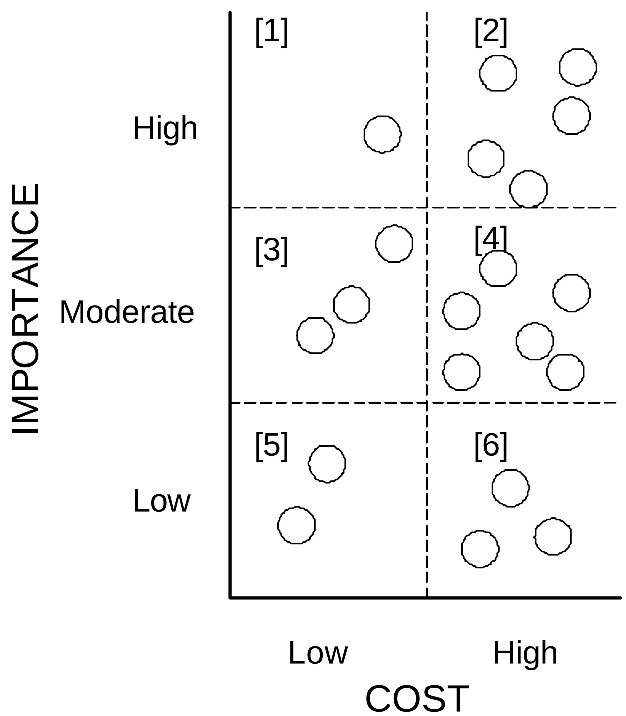


Figure 12. Graphical Representation of Problems for Comparing Cost and Importance [From Hix and Hartson, 1993].

Start with problems at the top of the list as candidates for priority. For example, look at some of the high-

importance/ high-cost problems perceived to be so critical that they must be fixed despite their high cost. Typically, it helps to prepare three separate lists: one for those problems that definitely are going to be addressed, one for those to be addressed if there is time, and one for those that are tabled for now (and perhaps for always). Also try to maintain some priority order within these lists, so that in the event that you run out of time before solving the problems you expected to fix, you have, at least, been attacking them in what you believe to be the best order [Hix and Hartson, 1993].

H. REDESIGNING AND IMPLEMENTING THE REVISED INTERFACE

Much of the work for this final phase of formative evaluation has already been done, when design solutions for each of the observed problems were proposed. At this point, developers need only to update the appropriate design documentation to reflect the decisions, and to resolve any conflicts or inconsistencies in the interaction design that might have resulted from the decisions. In addition, developers should make sure that the design is still a cohesive, comprehensive design that has not been affected, say, at a global level by any small detailed design decisions made to address specific low-level problems. It is then possible to proceed with confidence to implement the chosen design decisions. This is, of course, when developers realize the full benefits of formative evaluation, moving out of the current cycle of evaluation, and connecting back into the star life cycle, specifically into the subsequent cycle of (re)design, (re)implementation, and (re)evaluation [Hix and Hartson, 1993].

III. PROBLEM IMPLEMENTATION

A. OVERVIEW

The structure and use of the taxonomy is discussed in Chapter II — Problem Definition in detail. We saw that the structure is in non-linear form and end users need a very navigable application.

VE devices and methodologies have not matured yet and they are still in development phase. So the content of the taxonomy will need to be revised in the near future, and some parts may be changed, removed or added. This forces our application to be dynamic.

In order to support the content of taxonomy, some features may be improved like adding movie clips, figures etc. This will help to understand the context much better than simple text version.

The taxonomy was constructed in 1997 and nothing had been added to it since then. Most of the users lack of the usage of this valuable information source. In order to meet these needs, an implementation of WWW version seems to be a good candidate which is supported by dynamic database.

B. SOFTWARE AND DATABASE IMPLEMENTATION

The tools that are used for implementation are Macromedia Dreamweaver 6.0 Education Version, Macromedia Fireworks 6.0 Education Version and Microsoft Access. At first Extensible Markup Language (XML) based tools also were considered for implementation purposes but later we decided on the Macromedia and Microsoft Access. We decided that the learning curve of XML supported tools are too high and these tools need too much hand manipulation.

On the other hand, Macromedia and Microsoft Access are not so hard to learn and they can generate the code for you. The tutorials are good and can be finished in short time.

The guideline tables and references stored in Microsoft Access database. The information is retrieved from the database using Active Server Pages (ASP). The details of database structure will be discussed in next paragraphs.

Context-driven discussion sections converted to Hyper Text Markup Language (HTML) format. The links between guidelines/references and context-driven discussion also stored in the database.

The guidelines and references information are stored in five Access tables. These tables are:

		SECTION_NO	SECTION_NAME
	+	1	Users and User Tasks in VEs
	+	2	The Virtual Model
	+	3	VE User Interface Input Mechanisms
	+	4	VE User Interface Presentation Components
▶		0	

Table 3. CHAPTER5 Table — Section Information

CHAPTER5 table contains the names of the sections. These section names are the big box names in Figure 5. Primary Key (PK) is SECTION_NO field.

		SECTION_NO	TABLE_NO	TABLE_NAME
	+	1	1	VE Users
	+	1	2	VE User Tasks
	+	1	3	Navigation and Locomotion
	+	1	4	Object Selection
	+	1	5	Object Manipulation
▶	+	2	6	User Presentation and Representation
	+	2	7	VE Agent Presentation and Representation
	+	2	8	Virtual Surrounding and Setting
	+	2	9	VE System and Application Information
	+	3	10	VE User Interface Input Mechanisms in General
	+	3	11	Tracking User Location and Orientation
	+	3	12	Devices Supporting "Natural" Locomotion
	+	3	13	Data Gloves and Gesture Recognition
	+	3	14	Magic Wands, Flying Mice, SpaceBalls, and Real-World P
	+	3	15	Speech Recognition and Natural Languauge Input
	+	4	16	Visual Feedback — Graphical Presentation
	+	4	17	Aural Feedback — Acoustic Presentation
	+	4	18	Haptic Feedback — Force and Tactile Presentation
	+	4	19	Environmental Feedback and Other Presentation
*		0	0	

Table 4. CH5_SECTION_TABLE Table — Contains Table Names for Sections

CH5_SECTION_TABLE Table contains table names for each section. These names are the names of small boxes in Figure 5. Primary Key is TABLE_NO field.

	TABLE_NO	RULE_NO	LABEL	RULE	LINK_DESCRIPTION
+	1	1	Users1	Take into account users experience (i.e., support both expert and novice users)	/Web/Chapter6/Chapter6.htm#Users1
+	1	2	Users2	Support users with varying degrees of domain knowledge	/Web/Chapter6/Chapter6.htm#Users2
+	1	3	Users3	Take into account users' technical aptitudes (e.g., orientation, spatial visualization, and spatial memory)	/Web/Chapter6/Chapter6.htm#Users3
+	1	4	Users4	Support both right and left-handed users (e.g., through devices)	/Web/Chapter6/Chapter6.htm#Users4
+	1	5	Users5	Accommodate natural, unforced interaction for users of varied age, gender, stature, and size	/Web/Chapter6/Chapter6.htm#Users5
+	2	1	Tasks1	Take into account the number and locations of potential users	/Web/Chapter6/Chapter6.htm#Tasks1
+	2	2	Tasks2	When designinig collaborative VEs, support social interactions among users (e.g., group communication, role-play, informal interaction)	/Web/Chapter6/Chapter6.htm#Tasks2
+	2	3	Tasks3	In collaborative VEs, support cooperative task performance (e.g., facilitate social organization, construction, and execution of plans)	/Web/Chapter6/Chapter6.htm#Tasks3
+	2	4	Tasks4	Provide awareness-based information for competitive task performance	/Web/Chapter6/Chapter6.htm#Tasks4
+	2	5	Tasks5	Support concurrent task execution	/Web/Chapter6/Chapter6.htm#Tasks5
+	2	6	Tasks6	Design interaction mechanisms and methods to support user performance of serial tasks and task sequences	/Web/Chapter6/Chapter6.htm#Tasks6
+	2	7	Tasks7	Provides stepwise, subtask refinement including the ability to undo	/Web/Chapter6/Chapter6.htm#Tasks7

Table 5. A Portion of CH5_TABLES Table — Contains Information for Each Guideline Table.

CH5_TABLES Table contains all guidelines and related information for each guideline. Reference information for each guideline is stored in another table. Reference is optional for guidelines. Also DESCRIPTION_LINK field added to this table in order to navigate the context-driven discussion documents from guideline tables. This is a simple link which shows the exact place of the guideline in the context-driven discussion document. Primary Key is TABLE_NO and RULE_NO together.

TABLE_NO	RULE_NO	REFERENCE_NO
1	1	38
1	2	38
1	3	38
1	3	33
1	3	117
1	3	115
1	5	130
1	5	12
1	5	70
2	2	140
2	3	79
2	3	8
3	1	34
3	2	75
3	2	33

Table 6. A Portion of RULE_REFERENCES Table — Contains Reference Info for Each Guideline

RULE_REFERENCES Table contains reference information for each guideline. Guidelines may have reference information or not. If a guideline has reference(s) then, this table makes connection between CH5_TABLES and REFERENCES.

	REFERENCE_NC	REF_ABBRIVATION	REFERENCE_NAME
+	1	[Alusi et al, 1997]	Alusi, G., Tan, A. C., Linney, A. D., Raoof, K., and Wright, A. (1997). Three dimensional tracking with ultrasound for augmented reality applications in skull base surgery. In CVRMed-MRCAS '97. First Joint
+	2	[Applewhite, 1991]	Applewhite, H. (1991). Position tracking in virtual reality. In Proceedings of Virtual Reality '93. Beyond the Vision: The Technology, Research, and Business of Virtual Reality, pages 18, Westport, CT
+	3	[Ascension Technology Corporation ,1997]	Ascension Technology Corporation (1997). Burlington, VT, USA (http://www.ascension-tech.com/).
+	4	[Badler,et al 1986]	Badler, N., Manoochehri, K., and Baraff, D. (1986). Multi-dimensional input techniques and articulated figure positioning by multiple constraints. In Proceedings of the 1986 ACM Workshop on Interactive 3D
+	5	[Barfield and Danis, 1996]	Barfield, W. and Danis, E. (1996). Comments on the use of olfactory displays for virtual environments. Presence: Teleoperators and Virtual Environments, 5(1):109-121.
+	6	[Barfield et al., 1997]	Barfield, W., Hendrix, C., and Bystrom, K. (1997). Visualizing the structure of virtual objects using head tracked stereoscopic displays. In 1997 IEEE Virtual Reality Annual International Symposium Proceedings,
+	7	[Barfield et al., 1995]	Barfield, W., Zeltzer, D., Sheridan, T., and Slater, M. (1995). Presence and performance within virtual environments. In Virtual Environments and Advanced Interface Design, chapter 12, pages 473-513. Oxford University

Table 7. A Portion of REFERENCES Table — Contains Information about All References

REFERENCES Table contains information about all references. Primary Key is REFERENCE_NO field.

After creating these tables, we linked these tables with relationships. For this purpose, we used Entity Relationship Diagram (ERD) in Microsoft Access.

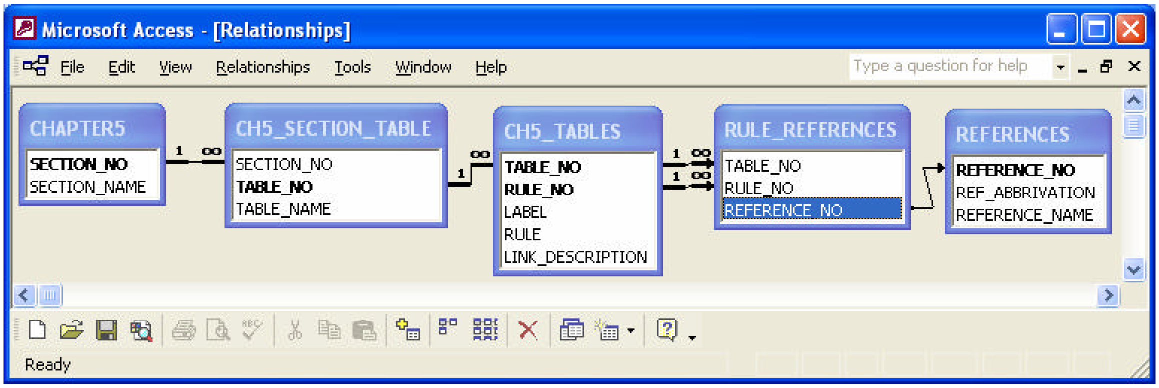


Figure 13. Entity Relationship Diagram (ERD) in Microsoft Access.

As you may notice in Figure 13, *bold field* names are Primary Keys. These are SECTION_NO, TABLE_NO, TABLE_NO&RULE_NO and REFERENCE_NO. All the field names in the tables can be read easily.

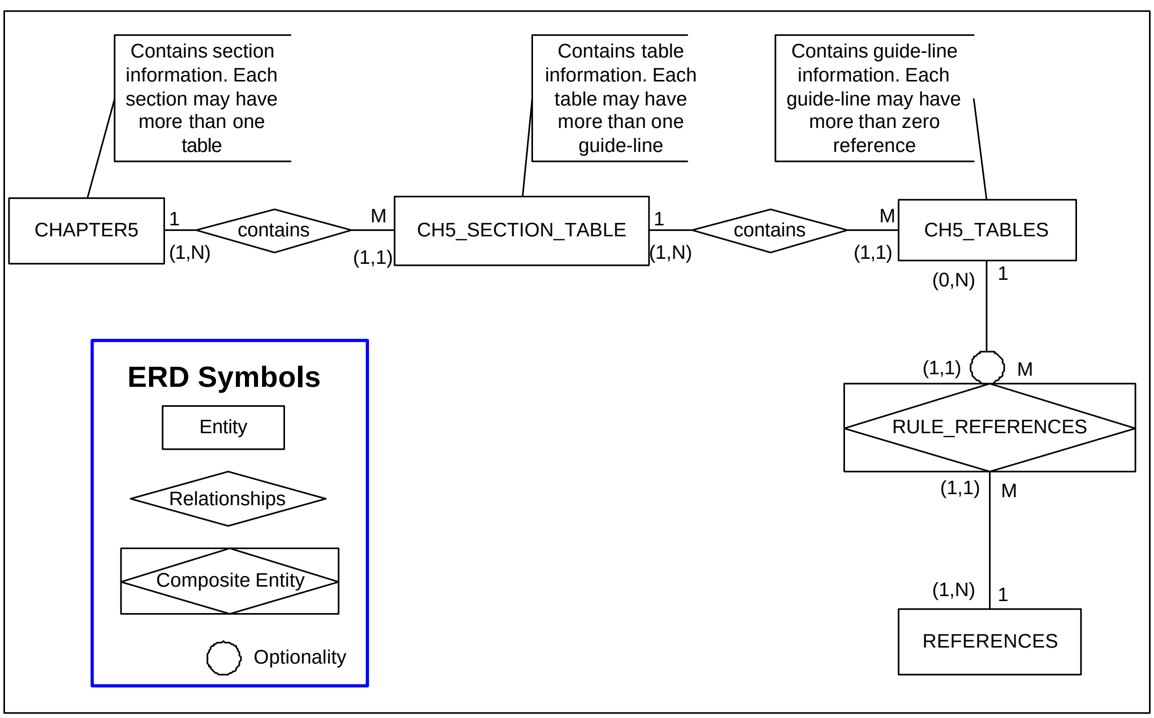


Figure 14. Entity Relationship Diagram(ERD) of Database in Detail

In Figure 13 and 14, Entity Relationship Diagram (ERD) is presented. The information is organized with these relationships. CHAPTER5 contains the information about sections. Each section may have more than one table. Because of this, relationship between CHAPTER5 entity and CH5_SECTION_TABLE entity is one-to-many (1:M). This relationship is the same between CH5_SECTION_TABLE entity and CH5_TABLES entity. Each table may have more than one rule (guideline). Each guideline may have more than zero references. As you can see reference is optional for guidelines. Each reference can be included by more than one guideline. For more information about ERDs see [Rob and Semann, 2000].

After building of taxonomy database, retrieving necessary information is handled by queries in ASP.

C. USER INTERFACE DESIGN

After discussing the structure of the taxonomy, it is time to talk about user interface design. In user interface design, we tried to be parallel to the taxonomy structure.

Implementation of navigable, readable and dynamic interface was the biggest handicap.

First a *prototype* was designed in Front Page. You can see menu structure in Figure 15 and the graphical representation of this prototype in Figure 16 and 17. This design was very close to the paper form of the taxonomy. In paper form, specific usability suggestions (guidelines) consist of a chapter. Explanations about these guidelines (context-driven discussion) divided into four chapters. These four chapters are the main titles of usability characteristics (see Figure 5 shaded boxes). We thought each of these chapters as a navigation bar (see Figure 15). After that we draw two sample pages in Front Page. Even though these sample pages are not active, they will be used to help to understand the visual design of the site. In later parts of the design, we thought that we may need to add extra navigation buttons to the navigation bar. In this case,

navigation bar is going to cover a bunch of buttons that may cause screen to seem messy. This was our first proposal.

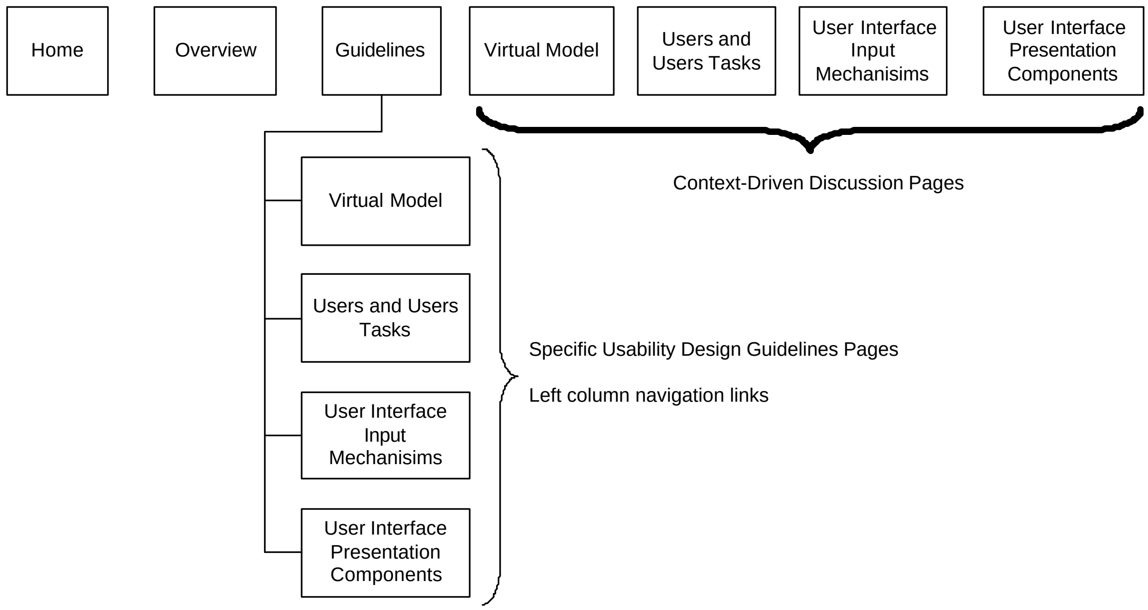


Figure 15. Prototype I Menu Structure

As you can see from the Figure 15 and 16, all the design guideline tables linked to a single button — *Guidelines*. When you select that button, a long list of guidelines table titles offered on the left column. You can visit whichever table you want by selecting the table title. Table contents will be on the right column. At first glance, this structure seemed to have problems, because there will be a long list of tables on the left column. This will also show the screen usages unbalanced and messy.

For context-driven discussion, we used four navigation bars. When you select one of these, the sub-titles of this chapter will be offered on the left column. You can visit any of these sub-titles by selecting this sub-title. The content of this sub-title will be on the right column (see Figure 17). It seemed that for each sub-title, we have to write a document/file and show that document/file in the right column. On the other hand, people have a habit and tendency to read the papers on the web. The design of papers is not like this. Usually the papers are not divided into pages, on the contrary, they are kept as a whole. The sub-

titles are written on the top and navigation links are attached to these sub-titles. When reader wants to jump to that part, it is very easy — just click that sub-title and you are there (see Figure 34).

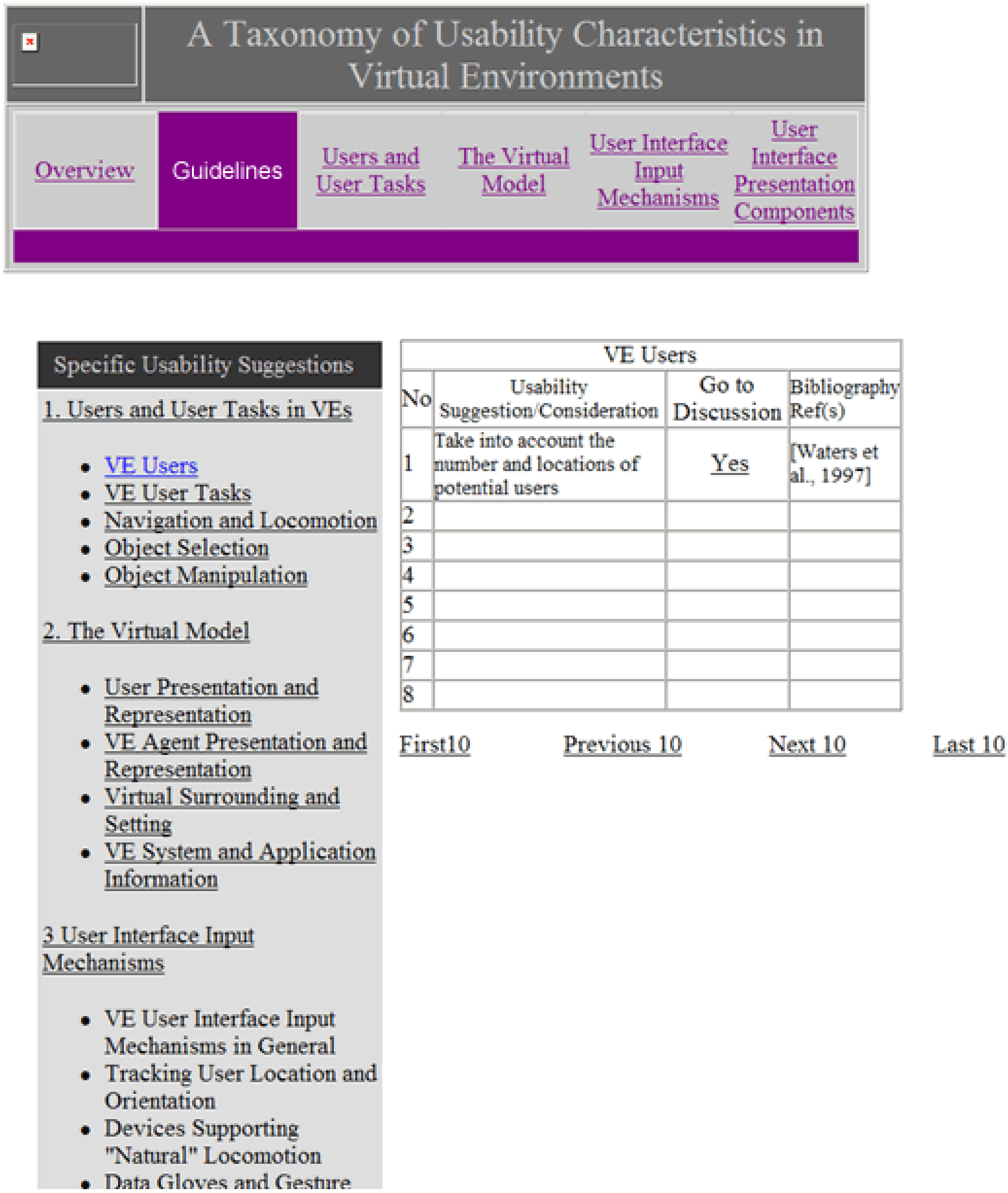


Figure 16. Prototype I Design Sample 1

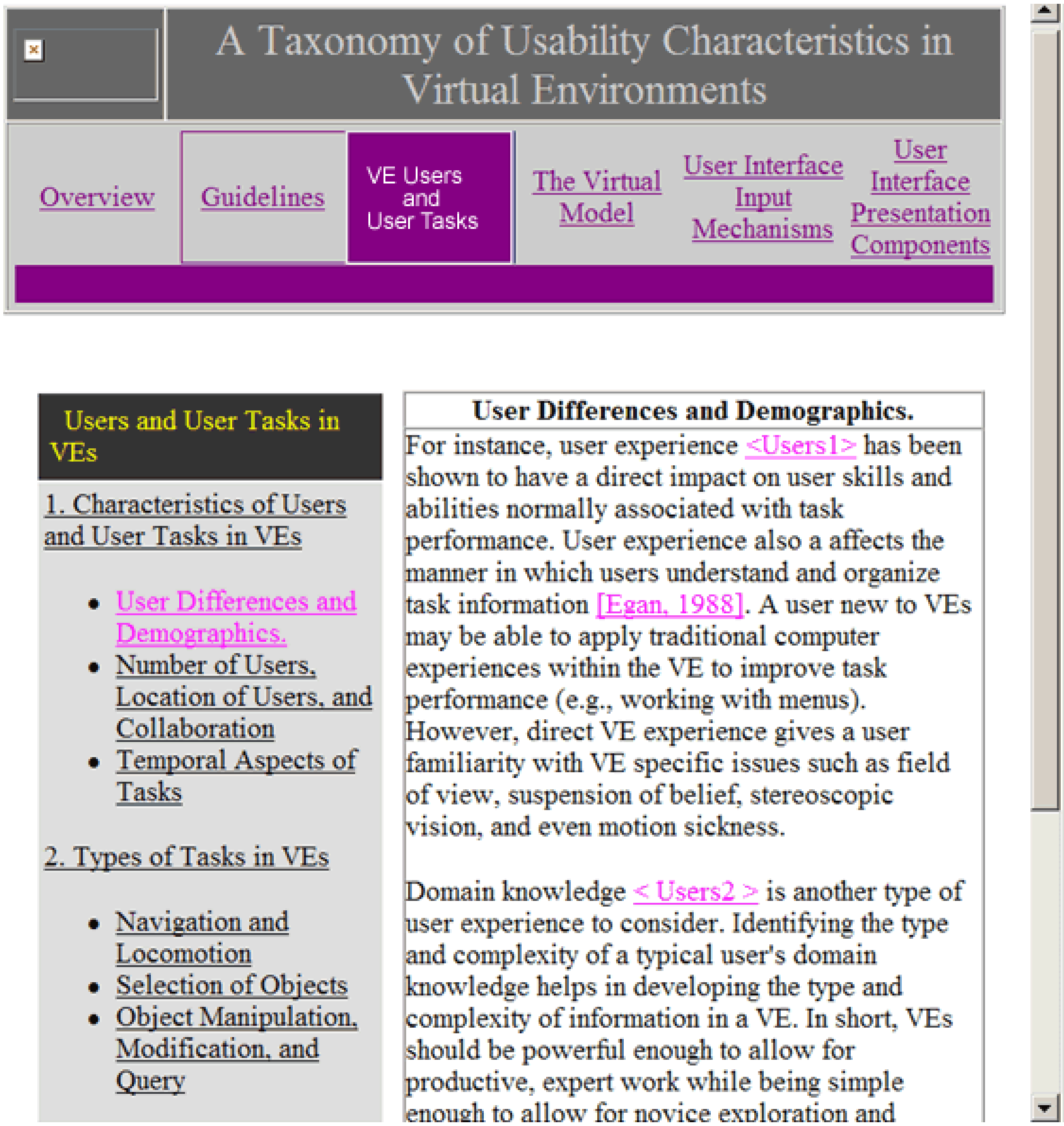


Figure 17. Prototype I Design Sample 2

After this point, we examined some well-known web pages to get an idea about how the navigation and layout are handled in these web pages. We liked the combination of navigation bar and tabbed pane design. We saw this design in Microsoft Hotmail and thought that we can use the same layout. You can see the menu structure of this design in Figure 18.

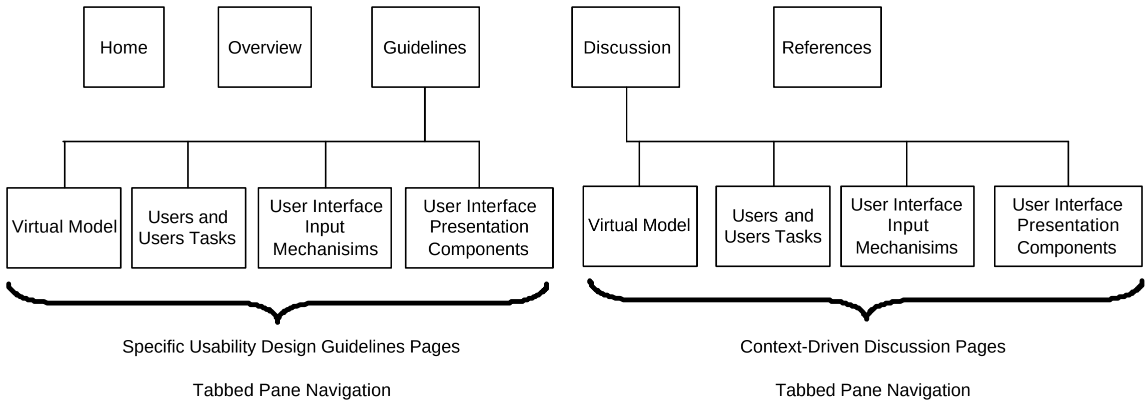


Figure 18. Prototype II Menu Structure

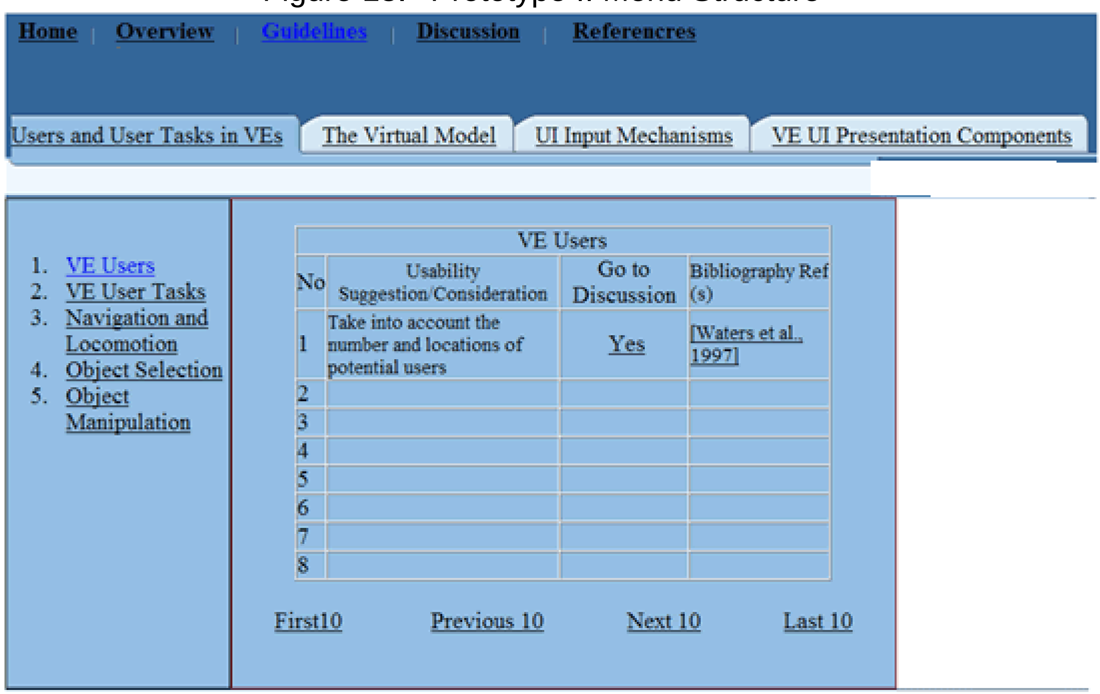


Figure 19. Prototype II Design Sample

We draw a sample page graphically in Front Page to see the layout (see Figure 19). At first glance, we thought that this layout would be good for guidelines and later decided to use same layout for context-driven discussion. Because they have the same layout structure, only the content is different. This will also decrease the number of navigation buttons in the navigation bar. Four context-driven discussion navigation buttons will merged under one button — *Discussion*. When you select this button, you will face the same tabbed pane that used for guidelines.

We added *acronyms* to this design and changed the *discussion* to *descriptions*. *Descriptions* navigation button is much more descriptive than *discussion* navigation button for context-driven discussion (see Figure 20).

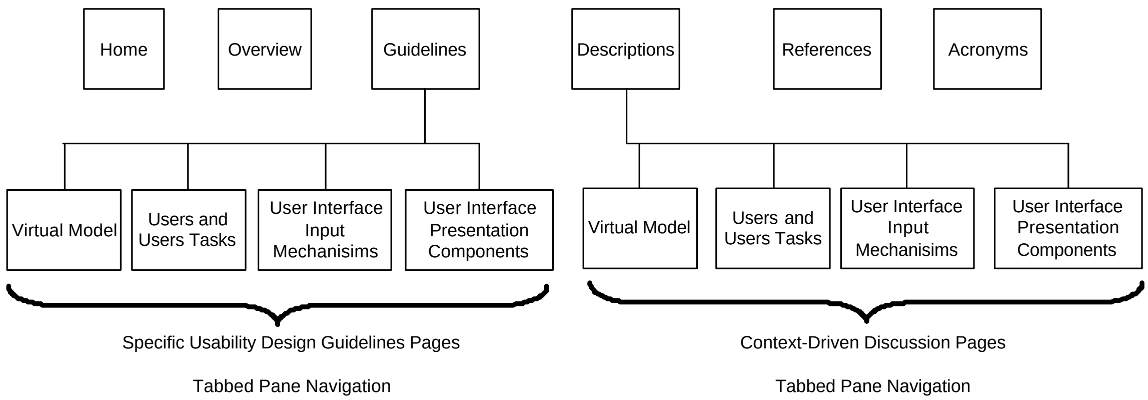


Figure 20. Menu Structure of Final Prototype

These prototypes are shown to a couple of users and they preferred the second one as we expected.

After this point, we focused on how to implement *Guidelines*, *Descriptions* and *References* pages.

As we mentioned in previous section, guidelines information was converted to Microsoft Access database. We preferred to use ASPs to retrieve guidelines information and present them in table structure. During implementation, there have been changes on column fields of guidelines table prototype. We returned to the original table structure and added links to the *labels*. When this link selected, it takes you directly to the related part of context-driven discussion page (see Figure 21). So we removed page numbers from the tables. We also put links to the references inside the tables. When you select that link, a window opens and shows the information about that reference (see Figure 22).

We converted context-driven discussion pages to four HTML pages. We placed subsections at the top and linked them to the related sub-sections. When

these sub-sections selected, it took you to the related sub-section. We also placed *anchors* and named them with the same name of *labels*. With the help of these anchors, we can find the place of guidelines and relate/link these parts with *Guidelines* tables.

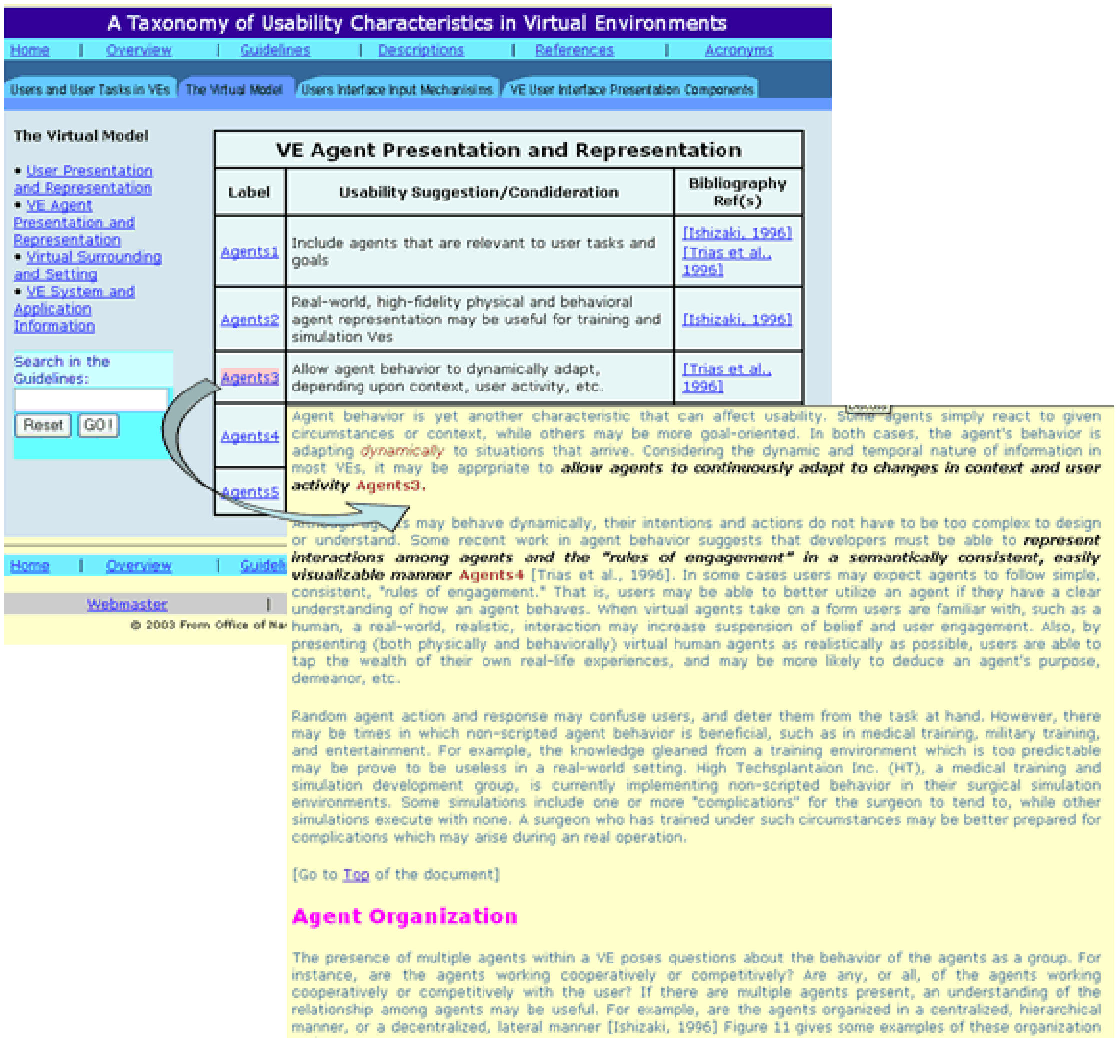


Figure 21. Accessing Descriptions Page via Guidelines Page

Now you can see the final design and layout of the *Guidelines* page in Figure 21 or 22. Look at the *navigation bar, tabbed pane, table titles* and *guideline table* layout. We tried all pages to be seen balanced — not to overweigh the information in any part of the screen.

In Figure 21, we showed how you can access the context-driven discussion of *Agents3 labeled guideline*. When you select *Agents3* label within the guideline table, you immediately reach the part of context-driven discussion that this guideline is discussed in detail.

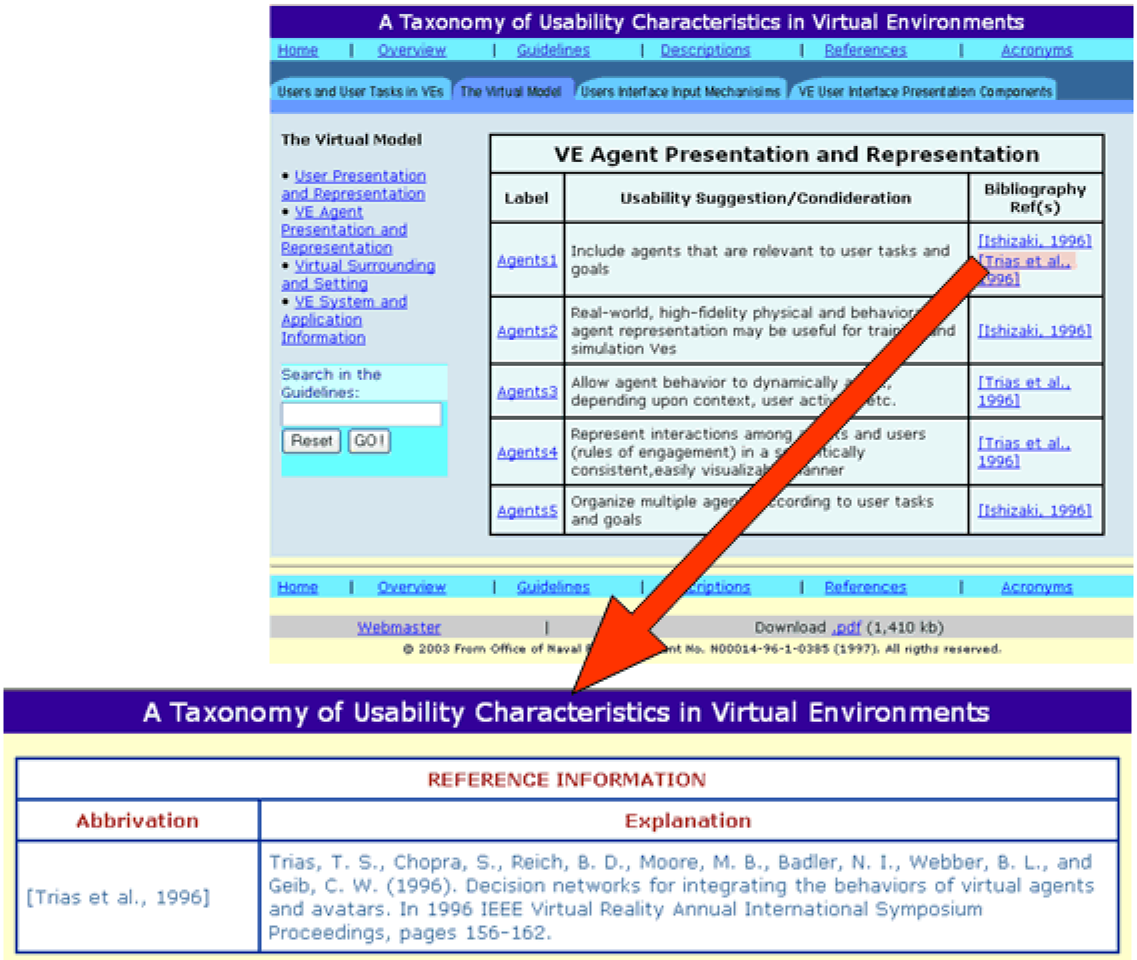


Figure 22. Accessing Reference Info via Guidelines Page

In Figure 22, you see how you can access the detailed reference information by selecting the related reference abbreviation. After clicking the reference abbreviation a window opens and shows the detailed information about this reference. After reading this information, you can easily find this reference if you need more information.

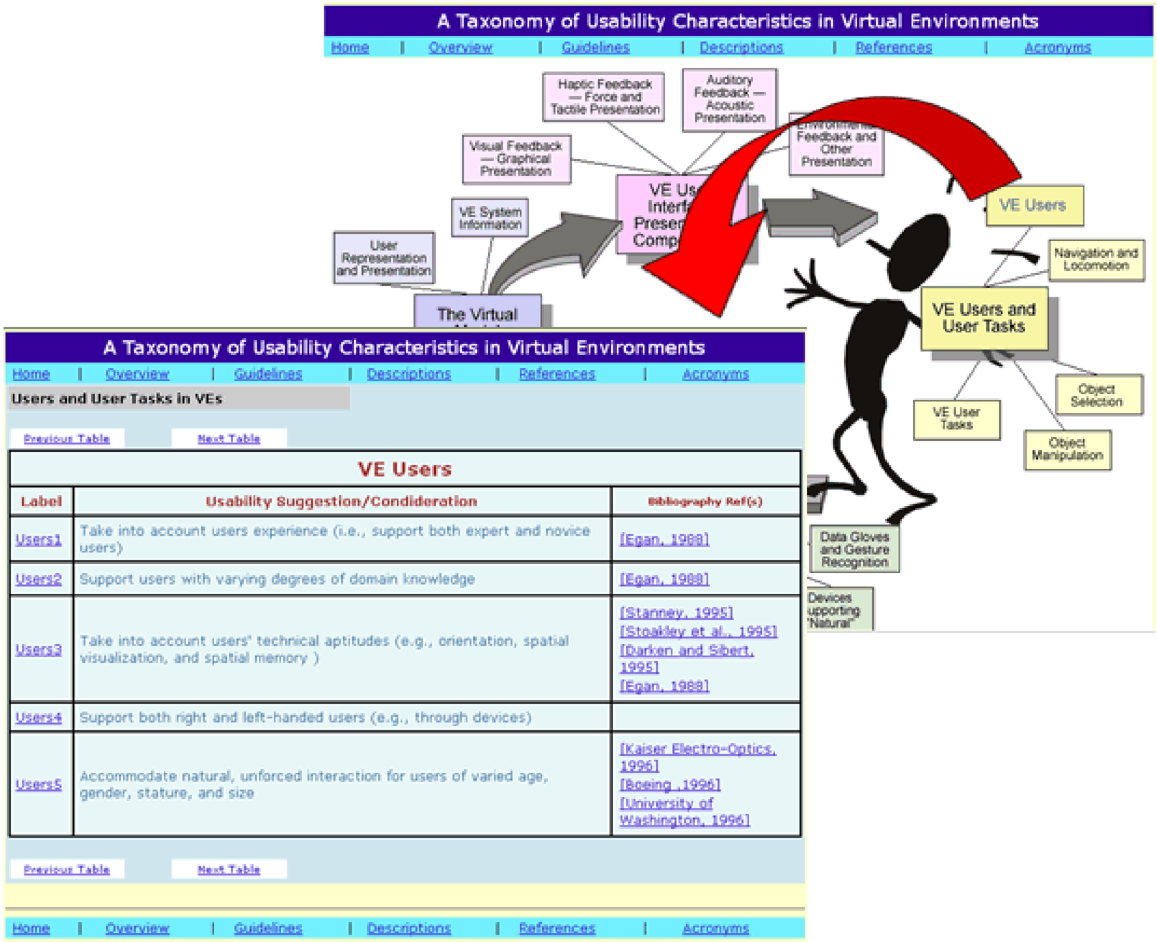


Figure 23. Accessing Guidelines via Overview Figure from Home Page

Next step was to build the *Home* page. We thought that an overview figure based simple page would be a good candidate for *Home* page. We iteratively improved this page (see Figure 30).

When you examine the overview figure, it has a circular structure. We linked the related *guidelines* table to each of text boxes in overview figure. When you select any of these boxes, you reach the related design guidelines table — top-down approach (see Figure 23**)**. These tables also have a circular structure like overview figure. You can navigate each guidelines table by using *next table* or *previous table* links. So the structure of *these guidelines* tables and *overview figure* is consistent. Therefore you have two choices to use the *guidelines* tables. One is to use via the *Home* page figure, the other is to use via the navigation bar

— *Guidelines*. The table content structure is the same for both navigation designs. Later, search engine added to the *Guidelines* page in order to search for special topic in the guidelines.

References are also an important source for usability characteristics in VEs. Therefore, we added *References* page to the web site. The design was very simple. It was consists of three columns — order no, abbreviation and detailed information about references. The navigation was to see the references five by five. In iterative cycle, we removed *order no* from table because references were already sorted alphabetically. An important feature also added to this page later which was to build search engine for references.

Acronyms are also added later to the site as we feel that users will need them. In usability design we saw that adding this page was a good idea. We did not change this page in iterative design cycle.

During initial design phase and iterative usability test we followed some usability guidelines. These guidelines helped us much to improve the user interaction with web site:

- Know the user — we considered user characteristics such as they know basic computer usage, general VE terminology etc.
- Prevent user error
- Optimize user operations — we try to increase efficiency as much as possible. Especially for navigation, *frames* are used and we got good results.
- Keep the locus of control with user — User in charge rather than computer.
- Give the user a mental model of the system, based on user tasks — we thought that the best mental model of the taxonomy is summarized in the overall figure (see Figure 5) and used this figure

in many places. Inside long scrolling, this picture showed where you are to prevent the panic that I am lost.

- Be consistent — to solve this, we used the same style sheet whenever it is possible. So the font, background, table, layout, headers… remained the same for all pages.
- Keep it simple — we try to keep the interface simple as much as we can.
- Try to minimize short term memory
- Let the user recognize rather than having to recall, whenever feasible.
- Use cognitive directness — again, the overall picture claimed this.
- Make user actions easily reversible — main navigation bar supported this need.
- Get the user attention judiciously — the first implementation of the overall figure in Home page did not offer what we expected. Some users perceived it as static figure, in fact it was dynamic — there was navigation links on the text boxes. Later, color and swap image behavior added as attention grabbers.
- Maintain display inertia — Templates was a good solution.
- Organize the screen to manage complexity

During the implementation of user interface, a formative usability analysis approach is conducted. This methodology will be discussed in next Chapter — Methodology.

D. ADDING TO THE TAXONOMY

One of the goals in this study is to make the taxonomy dynamic in order to expand and update contents of it. As we mentioned before, taxonomy site has a database and several HTML pages. Some pages like guidelines are constructed dynamically at run time by retrieving the information from database.

In order to add information to taxonomy, you have two choices. First one is to add information to the database while the other is to add to the HTML pages — usually context driven discussion pages. According to the complexity of the information added, you may add to *both* the database *and* HTML pages.

We reviewed the taxonomy database structure in section B — software and database part of implementation. In this database, we stored information about *guidelines* and *references*. You may change this information very easily. The structure of database is in table forms. By looking at these tables, you can easily find what you are looking for. These tables are related with each other with ERDs. You can navigate these tables by starting with one table.

First we want to show you the top-to-bottom navigation approach. We will start from the table that stores section names and navigate downwards. Now look at Figure 24 which stores section names. You can start from which section you want and see or change information. In Figure 24 table, you see three columns. First column does not have any name and just shows + signs. When you clicked one of this signs, it expands and shows the table names related to that section (see Figure 25). When you apply the same action sequentially to the Figure 25 and Figure 26, you will get Figure 27. In Figure 27, you see a guidelines table and reference numbers of one of this guidelines. You saw that it is very easy to navigate between these pages as these pages are related with ERDs.

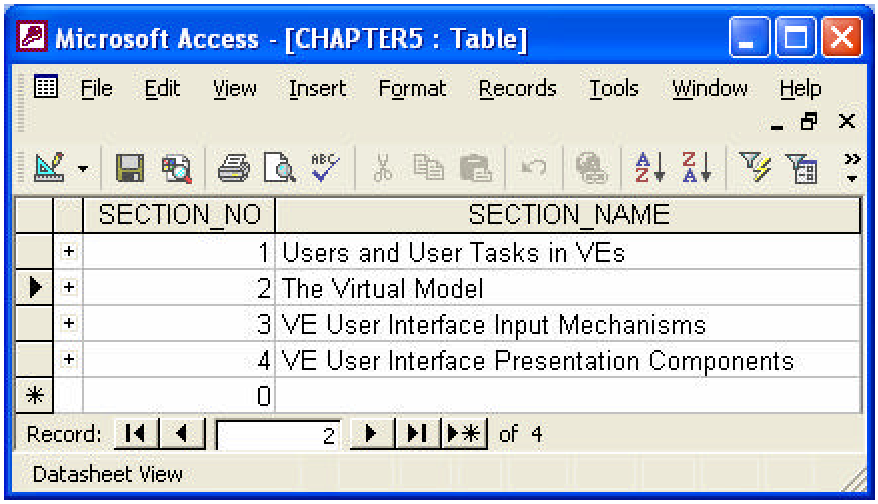


Figure 24. Table of Section Names

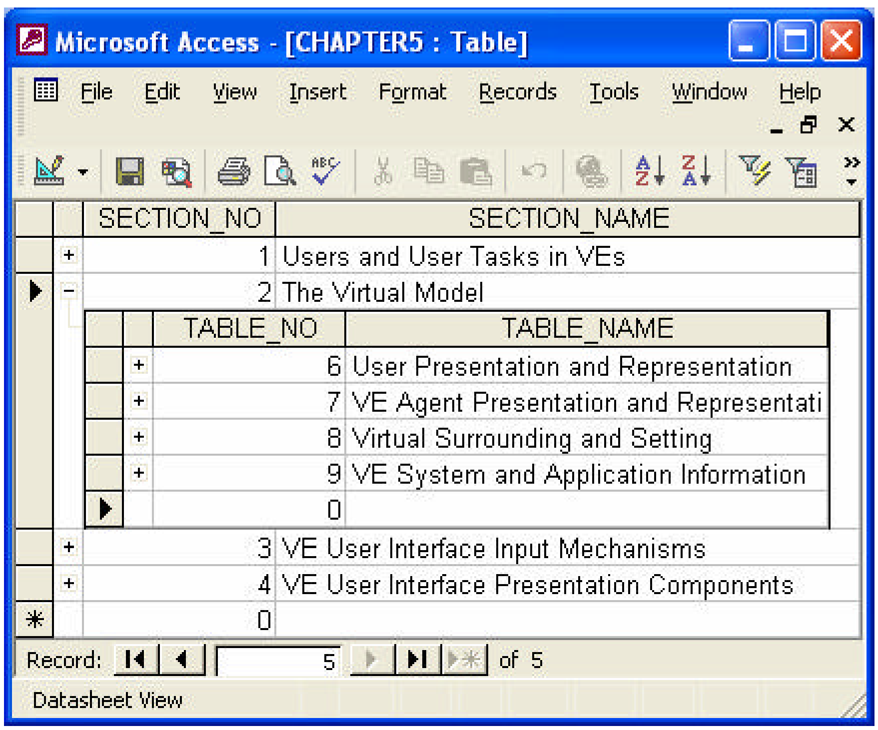


Figure 25. Tables of Virtual Model

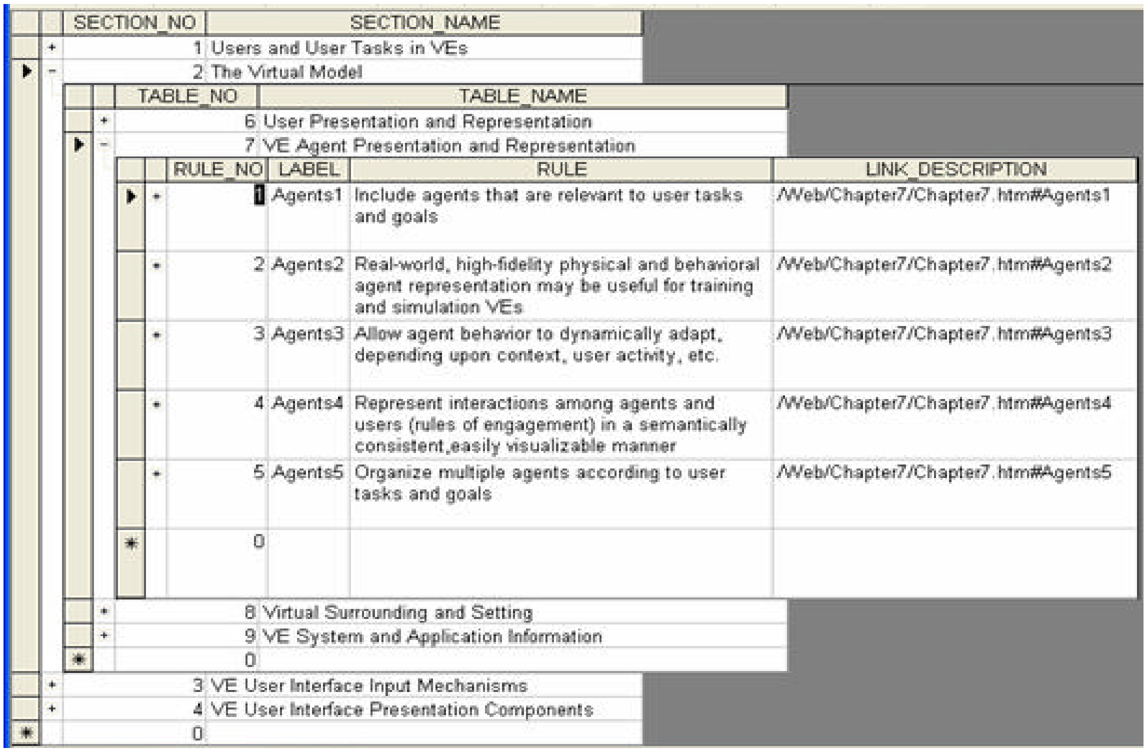


Figure 26. Guidelines of VE Agent Presentation and Representation

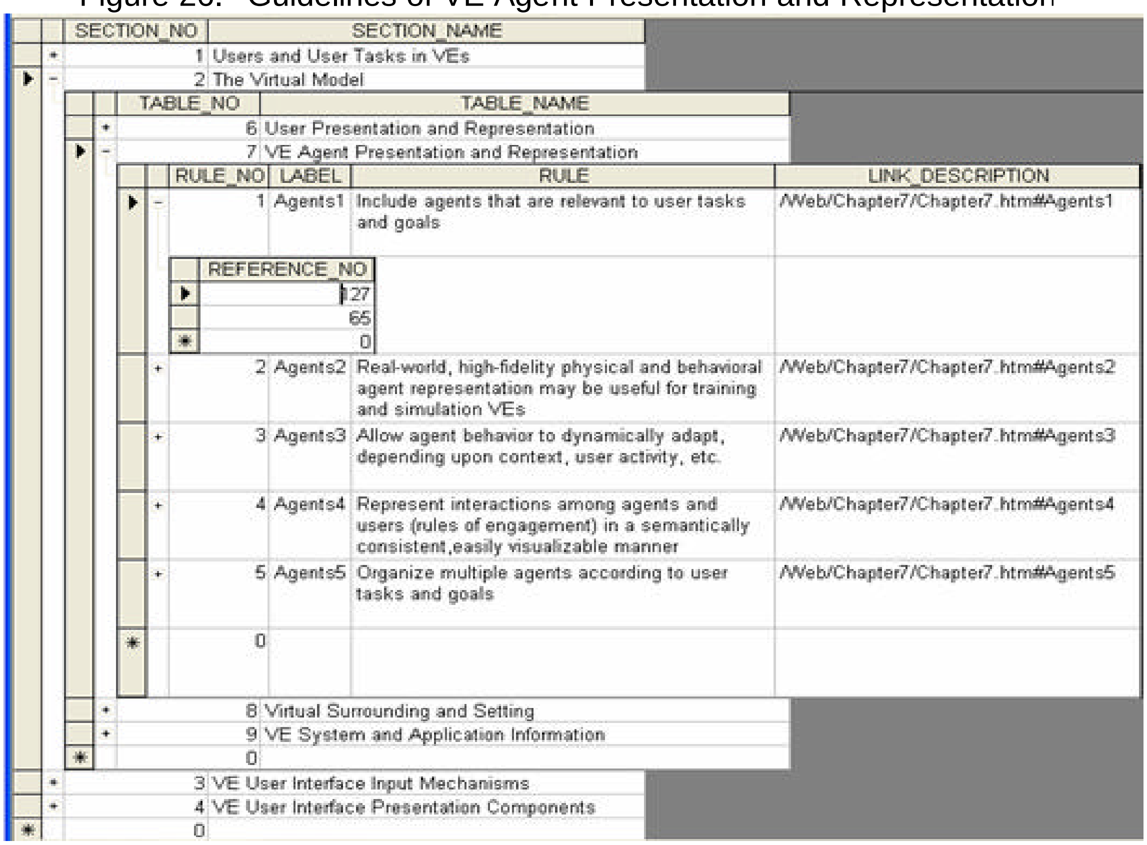
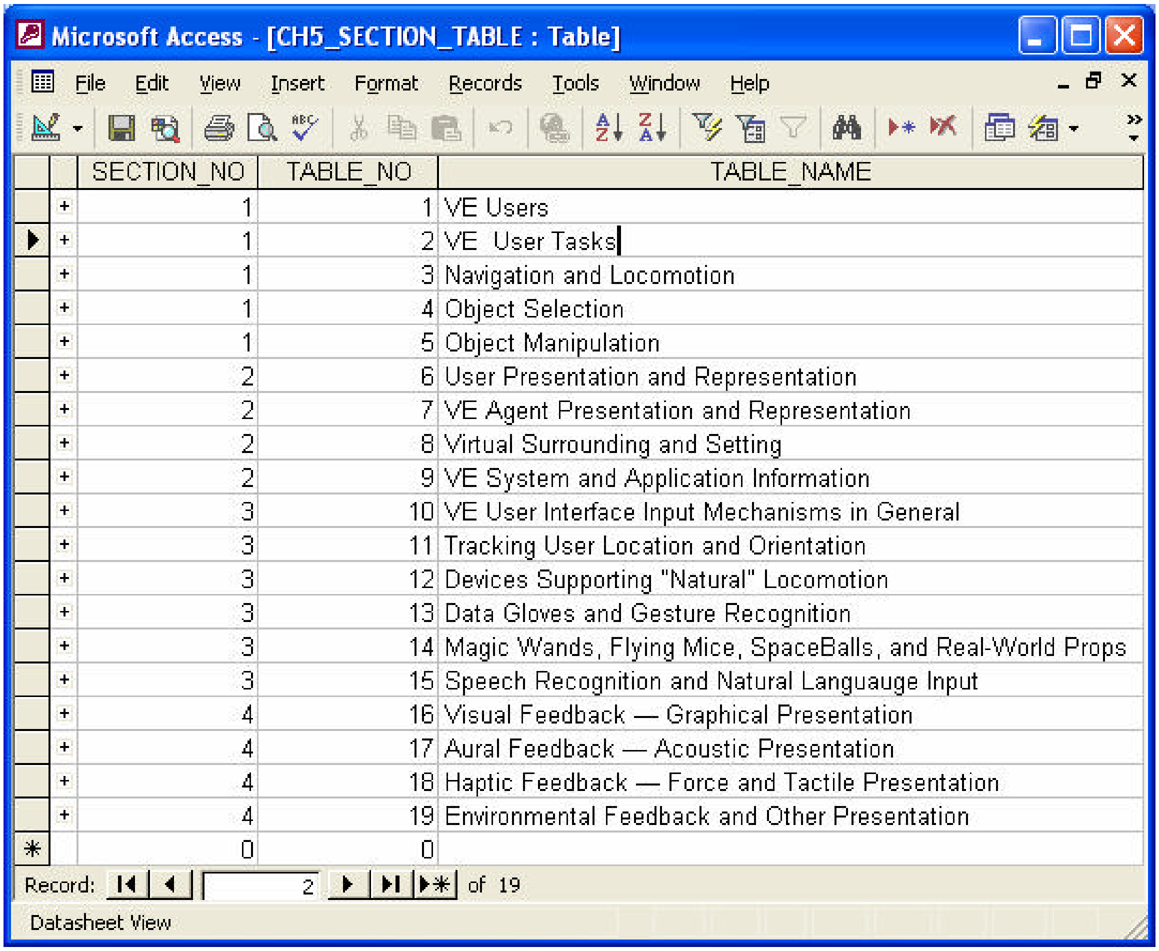


Figure 27. Reference Sample

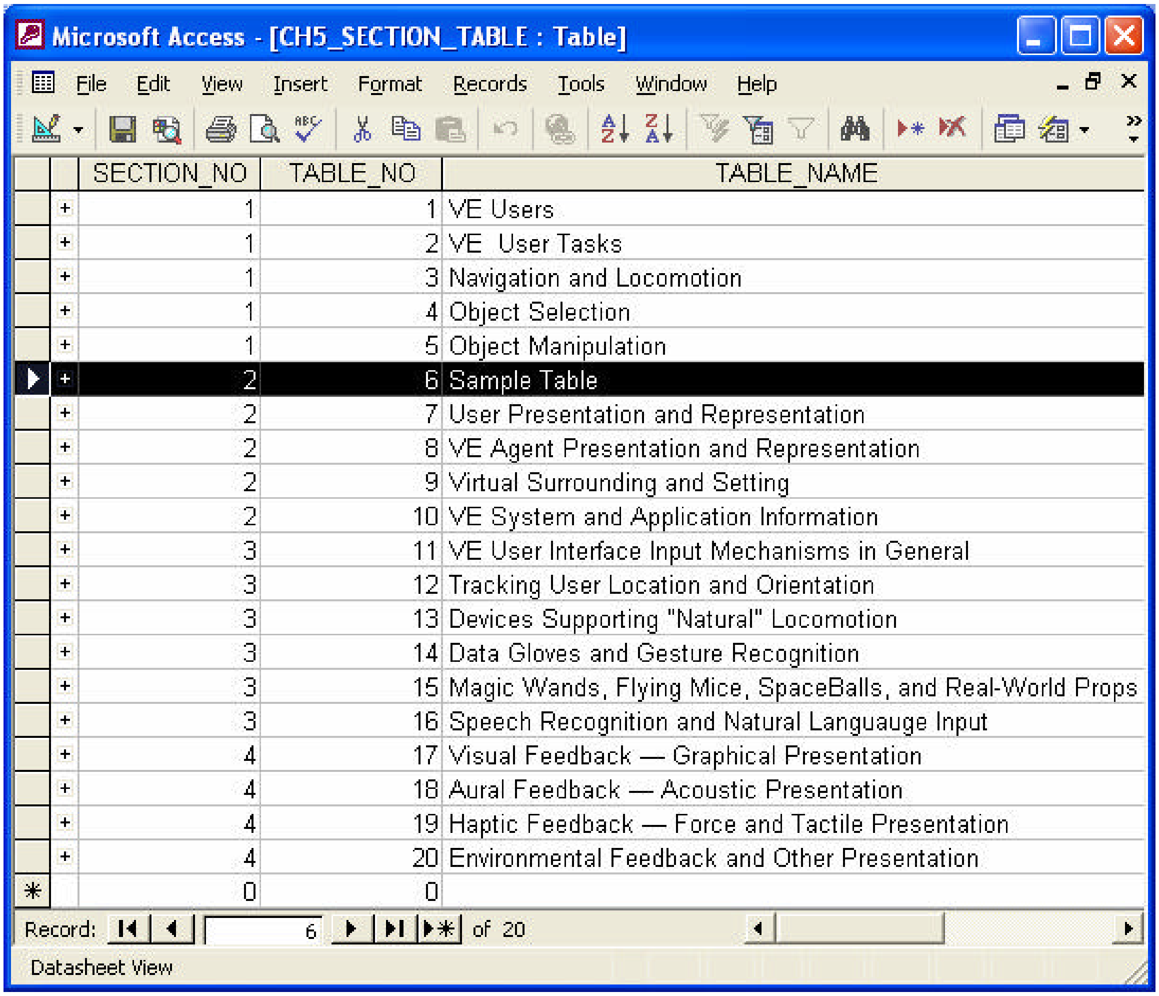
You can start from any table and can navigate between these tables. Let's give another example:



SECTION_NO	TABLE_NO	TABLE_NAME
1	1	VE Users
1	2	VE User Tasks
1	3	Navigation and Locomotion
1	4	Object Selection
1	5	Object Manipulation
2	6	User Presentation and Representation
2	7	VE Agent Presentation and Representation
2	8	Virtual Surrounding and Setting
2	9	VE System and Application Information
3	10	VE User Interface Input Mechanisms in General
3	11	Tracking User Location and Orientation
3	12	Devices Supporting "Natural" Locomotion
3	13	Data Gloves and Gesture Recognition
3	14	Magic Wands, Flying Mice, SpaceBalls, and Real-World Props
3	15	Speech Recognition and Natural Languauge Input
4	16	Visual Feedback — Graphical Presentation
4	17	Aural Feedback — Acoustic Presentation
4	18	Haptic Feedback — Force and Tactile Presentation
4	19	Environmental Feedback and Other Presentation
0	0	

Figure 28. All Guidelines Tables

As you see in Figure 28, you can start from here to navigate downwards. As you can guess, the values in the table cells can be edited. Likewise you can add new items by filling the values in the last raw of each table. For example, in Figure 28, you can add new design guideline table by filling in the last row of this table whose all values are 0s. But you have to correct some fields manually. If you want to keep TABLE_NO sequentially according to sections, then you must reorder the TABLE_NOs manually (see Figure 29). We added a sample table to section 2 and shifted TABLE_NOs thereafter.



SECTION_NO	TABLE_NO	TABLE_NAME
1	1	VE Users
1	2	VE User Tasks
1	3	Navigation and Locomotion
1	4	Object Selection
1	5	Object Manipulation
2	6	Sample Table
2	7	User Presentation and Representation
2	8	VE Agent Presentation and Representation
2	9	Virtual Surrounding and Setting
2	10	VE System and Application Information
3	11	VE User Interface Input Mechanisms in General
3	12	Tracking User Location and Orientation
3	13	Devices Supporting "Natural" Locomotion
3	14	Data Gloves and Gesture Recognition
3	15	Magic Wands, Flying Mice, SpaceBalls, and Real-World Props
3	16	Speech Recognition and Natural Languauge Input
4	17	Visual Feedback — Graphical Presentation
4	18	Aural Feedback — Acoustic Presentation
4	19	Haptic Feedback — Force and Tactile Presentation
4	20	Environmental Feedback and Other Presentation
0	0	

Figure 29. Adding a Sample Guideline Table

After adding the name of guideline table, you can fill in the information to the table cells. We filled information in this table for two guidelines in order to show you an example (see Figure 30). You can add, delete or edit any guideline in the tables as we did here. By updating the database, you automatically updated the web site also.

If you need to update the HTML files with regard to changes in database, you should follow the same styles in these files. While adding a new guideline, you must put an *anchor* with the name of the label of that guideline. If you want emphasize the words of guideline, these words must be *emphasized-strong*

(italic-bold) or *strong-emphasized (bold-italic)*. You must also write the label in *bold (strong)* form at the end of this guideline.

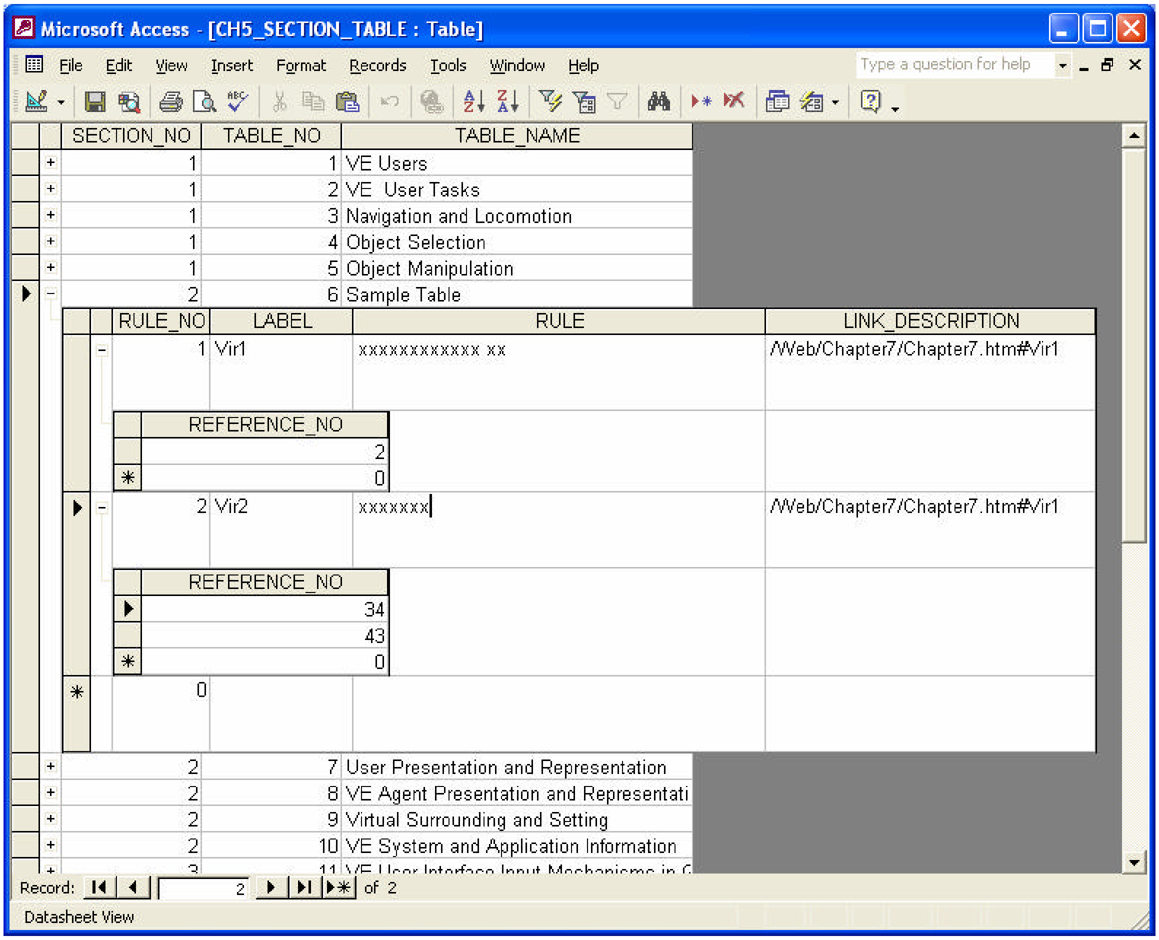


Figure 30. Guidelines Info Entry to Sample Table

With simple examples, you saw how to change/update the database. For HTML file changes, you can use some tools like Dreamweaver, Front Page or even a text editor.

E. ADDING A SAMPLE STUDY TO THE TAXONOMY

We wanted to add a sample study to the taxonomy to see if it's easy to do so or what kind of problems we are going to meet. Our study was about acquiring spatial knowledge with egocentric and exocentric views while navigating.

This taxonomy is in *Linnaean* taxonomy form. *Linnaean* taxonomies attempt to classify entities and groups in terms of their *essence*. There are no set

rules or procedures for how an entity is classified. This method involves significant subjective judgment as to the fundamental characteristics of an entity or group of entities. More importantly, the context in which an entity is to be classified has everything to do with the language used to describe it. An engineer might describe a glove device in terms of its components (e.g. fiber optics, stress sensors, etc.) while a physiologist might describe it in terms of the tasks for which it can be used (e.g. pointing, grasping, etc.). So it does not have a consistent set of rules for inserting new items [Cockayne and Darken, in press].

Cockayne and Darken [in press] were describing the problems as if we encountered during adding new study to the taxonomy. It was not so clear where our new study fits in the taxonomy and there were no rules to help us even though the structure and layout of this taxonomy was so well constructed and strong. There were three candidate places to add this study according to our judgments:

1. The Virtual Model -> Types of information present in virtual model -> VE system and application information -> Spatial information
2. Users and User Tasks in VEs -> Characteristics of Users and User Tasks in VEs -> User Differences and Demographics
3. Users and User Tasks in VEs -> Types of Tasks in VEs -> Navigation and Locomotion

This study may fit more than one place. It can be changed according to the judgments. As you can see, adding new studies to the taxonomies seems not so clear. We added our study to the Navigation and Locomotion part.

The people who are going to expand the taxonomy must know the structure and organization of the taxonomy very well. First they must find which part of taxonomy is suitable for their study. After finding the related section, they must refine their study. Because some parts of your study may be done by other researchers in the taxonomy already. If some part of study matches with some part of taxonomy, in this case, new study may be added as a new reference to

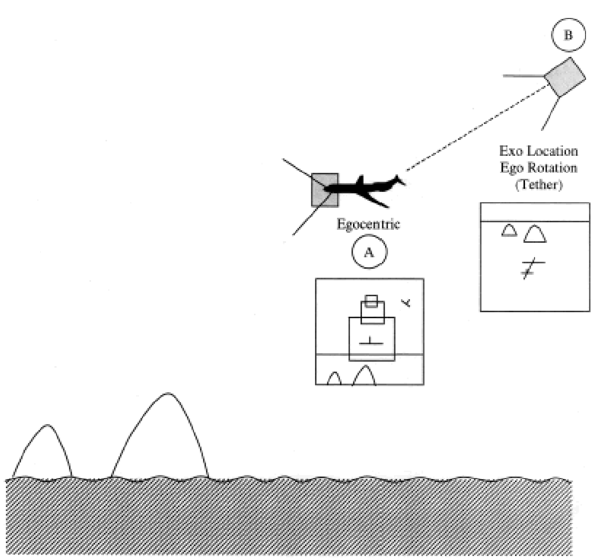
taxonomy. If new study is not included in the taxonomy then it's principles can be added to related guidelines table and a short explanation to the context-driven discussion section—*Descriptions* pages.

For adding purpose, we combined Wickens [2002] and Tokgoz [2002] studies. Scientific studies may be very long and cover lots of topics entirely and/or partially. We dealt with the egocentric and exocentric views of these studies. Egocentric and exocentric views were covered in Wickens' [2002] study partially while it was entirely in the Tokgoz' [2002] study.

As a first step, we extracted the parts from these studies that we will use to add to the taxonomy and then combined the results of these parts to extract a principle as follows:

> *Frame-of-reference* issue is another important factor to build spatial knowledge of an environment during navigation. If the environment is especially changing while navigating, it becomes more important. For example, in aviation and shiphandling, you have to consider the static objects and moving objects around you. Wickens [2002] [2] tries to propose the best cognitive model representation for aviators to help them understand situation awareness. The *frame-of-reference* issue concerns whether information should be presented from the pilot's frame of reference, an egocentric view of the airspace corresponding to what the pilot sees, or from an exocentric view of the airspace, stabilized to a world-centered frame. In this study he asks some questions to emphasize importance of frame-of-reference between egocentric ("inside out") and exocentric ("outside in") navigation: *Should the world rotate and translate around a fixed aircraft (egocentric), or should the aircraft rotate and translate on the display (exocentric)? Should the viewpoint show the pilot's forward view, or should it show the aircraft from above and behind?*
>
> The answers to these questions *depend on both the task and the user*. For example, several studies have found that flight control (tracking accuracy) is much better with an egocentric view (Figure 2, viewpoint A), but that noticing hazards in the airspace (referred to as Level 1 spatial awareness, or Level 1 SA) and understanding their general location (Level 2 SA) are better served by a more exocentric view (Figure 2[2], viewpoint B; Wickens, in press[2]). Other studies have compared two kinds of egocentric displays: moving-aircraft displays, which are consistent with a mental model that represents an aircraft moving in a fixed environment, and fixed-aircraft, moving-environment displays, which are more familiar to skilled pilots. These studies have revealed that novice pilots are better served by moving-aircraft displays, but that skilled pilots track equally well with the two kinds of displays [Previc and Ercoline, 1999[2]]. Tokgoz [2002] [2] did a study to compare the spatial knowledge acquisition by using egocentric and exocentric navigation metaphors by using an aircraft in a non-complex virtual environment desktop display. In this study, egocentric view is tethered at behind—the tail—and above the aircraft while exocentric view always looking towards north—fixed-aircraft, moving-environment display. In this study he found individual differences among participants when constructing cognitive map. The distance judgments of participants in exocentric navigation were better than egocentric navigation, but they did not differ significantly. They underestimated the distances. On the other hand direction estimations were not so bad. Out of nine participants, one participant estimated directions wrong in exocentric navigation while this number was three for egocentric navigation.As you can see, both the distance and direction estimations were better with exocentric navigation, in turn, better spatial knowledge.

This conclusion also do not contradicts with Wickens [in press], on the contrary, supports it. But on the other hand, evaluator observations and post experiment participant reviews suggested that the control of the aircraft in egocentric navigation was easier than exocentric navigation which supports Wickens [2002] [2] results. The viewing frustum in exocentric navigation was always looking towards north. Some objects near the aircraft—not in viewing the frustum— can not be seen easily. In order to overcome this problem, changing the direction of viewing frustum as in Figure 2 viewpoint B—tethering to the direction of aircraft at a fix distance— may be more beneficial. Therefore, ***use egocentric view when positions and orientations of objects are important relative to user(s) such as flight control (tracking accuracy) while exocentric view is preferable when global orientation of objects are important to accomplish the task(s) such as noticing hazards in the airspace, understanding general locations of objects*** **Nav5** [Wickens, 2002; Tokgoz, 2002] [2].



The gray box near each display represents the "camera view" relative to the large black aircraft. The most egocentric representation (viewpoint A) is from the pilot's eye point. It depicts a three-dimensional (3-D), forward-looking command flight path "tunnel" (represented by the three squares, which are "windows," receding in depth, to be flown through) and the aircraft's current location (represented by the large inverted *T*); the small inverted *T* shows the predicted location of the aircraft a few seconds in the future. The 3-D exocentric viewpoint (viewpoint B) depicts the airplane (shown by the lines in the middle of the display) from behind and above; the view maintains a constant distance behind the plane, as if "tethered" to it by a rope (represented as the dashed

Figure 2 [2]: Two representations of a pilot's airspace as the aircraft approaches two hills [A portion of figure from Wickens, 2002]

Later, we decided where to place this information in the taxonomy. This decision was subjective for us. An automation process may be needed while adding new studies to the taxonomy that may help researchers very much.

After finding the correct place in the taxonomy, we put principle/guideline *label* at the end of guideline and shifted the *figure* and related *label* numbers in the taxonomy. As you may already recognize, the guideline is highlighted by making the guideline font bold-italic (strong-emphasized).

[2] Note that figure number and references do not refer to this document rather it refers to the web-based version of taxonomy.

This guideline also added to the Access Database in the related guidelines table with its references.

The context we added to the taxonomy may be already added to the taxonomy. Adding these studies will be redundant. In this case, Wickens [2002] and Tokgoz [2002] studies may be added to the references part of that context in the taxonomy. It is another possibility.

IV. METHODOLOGY

A. DESIGN

The objective of this study is to evaluate the usability of the user interface of Hypermedia Representation of Taxonomy [Gabbard and Hix, 1997] and to recommend alternatives to improve user interface of this application.

The interface is evaluated by using the *formative usability evaluation*. We discussed formative usability evaluation in detail in Chapter II, but let's recall it briefly again.

The goal of *formative evaluation* is to assess, refine, and improve user interaction by iteratively placing representative users in task-based scenarios in order to identify usability problems, as well as to assess the design's ability to support user exploration, learning, and task performance [Hix and Hartson, 1993]. *Formative usability* evaluation is an observational evaluation method which ensures usability of interactive systems by including users early and continually throughout user interface development. The method relies heavily on usage context (e.g., user task, user motivation, etc.) as well as a solid understanding of human-computer interaction and, as such, requires the use of usability experts [Hix and Hartson, 1993].

While the *formative evaluation* process was initially intended to support iterative development of instructional materials, it has proven itself to be a useful tool for evaluation of traditional GUI interfaces.

The steps of a typical *formative evaluation* cycle begin with development of user task scenarios, and are specifically designed to exploit and explore all identified task, information, and work flows. Representative users perform these tasks as evaluators collect both qualitative and quantitative data. These data are then analyzed to identify user interaction components or features that both support and detract from user task performance. These observations are in turn

used to suggest user interaction design changes as well as formative evaluation scenario and observation (re)design [Hix and Gabbard, 2001].

The major steps of the evaluation will include the following [Hix and Hartson, 1993]:

- Developing the experiment
- Directing the evaluation session
- Generating and collecting the data
- Analyzing the data
- Drawing conclusions to form a resolution for each problem
- Redesigning and implementing the revised interface

B. USER ANALYSIS

This taxonomy is expected to be useful for VE researchers and developers, as well as funding agencies. Specifically, researchers and developers can get a breadth and depth overview of usability characteristics that are important to VEs, and can find guidance, via the extensive supplemental usability resources (guidelines, discussion, and references), for examining design questions for VE applications they are producing [Gabbard and Hix, 1998].

Thus, the expected user pool is as follows:

- VE researchers and developers,
- Funding agencies, and
- VE related Master/PhD. Students

As you can see from the above picture, it does make sense to assume that users know the general terminology of the VEs. They have common knowledge about how to use computers, web pages and window operations.

C. DEVELOPING THE EXPERIMENT

Experiment is developed with following four main activities:

- Selecting participants
- Developing tasks
- Determining protocol and procedures
- Pilot testing

1. Selecting Participants

While selecting the participants, it is very important to select the participants among correct user pool. Because your application will be evaluated with the help of these participants. If you choose wrong participants, your evaluation may probably not give expected user reactions — even though your data analysis with wrong participants analyzed correctly. Because you evaluated the application without real users. It is like comparing apples with oranges.

First, possible users of this application are analyzed as in section B. Thereafter, we tried to select a good participant sample out of user population.

We looked for the possible participants that we can easily find and decided that we are living with these people in School. So we selected the participants among Master/PhD. students in CS/MOVES department who were doing VE related work at Naval Postgraduate School (NPS).

We assumed that user profile was familiar with VE terminology, mouse use and basic computer skills. Nine participants involved in this study.

2. Developing Tasks

Developing tasks is very vital in usability engineering in order to find problematic areas. You must choose good *representative* and *benchmark* tasks which covers all the areas of application that you will evaluate.

Usually in usability evaluations, these tasks are written in a list and participants try to perform these tasks sequentially. Evaluator(s) collect(s) *qualitative* and *quantitative* data during this time. When we consider the structure

and purpose of taxonomy, it did not seem a good idea to list these tasks and expect participants to do these tasks. So we selected a natural way which is more appropriate to evaluate the taxonomy. In real life, we expect the same situation.

We thought a simple VE design *scenario* which contains main tasks for taxonomy. While participants try to design this scenario, we collected data. You can take a look at this scenario in Table 8.

When you examine the design scenario, you may guess tasks that participants should do. At first look we can list some of these tasks:

- Understand the goal of web site.
- Use overview figure in the Home page.
- Understand general usability characteristics of VEs.
- Look guidelines about a special topic.
- Apply these guidelines to suggested VE design.
- Look detailed information about a guideline.
- Find a special reference information.
- Represents grenades that fit for this scenario.
- Model the explosions.
- Represent user(s).
- Model selecting the grenade(s).
- Model manipulating the grenade(s)
- Model triggering the grenade
- Model throwing away of grenades
- Select a good model for this scenario (CAVE, HMD, etc.)

SCENARIO

You are given a duty to design a VE which has the following features:

The goal of the VE is to train the recruit soldiers how to use the grenades.

In this VE, soldiers will pull out the pin of the grenade and will throw it away towards the varying distance targets. After a certain time of pulling out the pin of the grenade, it will explode and damage the targets according to success of hit.

- The grenades will explode after a certain time,
- Targets may appear at varying distances
- Soldiers must be able to throw the grenades whichever distance they want. If the soldier applies more force while throwing the grenades, grenades must go further and vice versa.

These are some issues to help you think your model representation:

- Grenade representation
- Grenade display/tracking
- Targets
- Explosions
- User representation
- Selection/manipulation of grenades
- Hand/glove tracking
- Trigger the grenade

Table 8. VE Design Scenario

You can expand this list. We just listed some tasks to show you that in order to do these tasks you must use most of the features of web site. While using these features, we will find good and bad sides of this design.

So this study is evolved as scenario based formative usability evaluation.

3. Protocol and Procedures

Objective

In this study we wanted to see how taxonomy is affecting the VE designers' decisions. In order to test this, the participants will design VE scenario without taxonomy. After this step they will reconsider their design with help of taxonomy web site. We will see the difference between two designs and compare the effects of taxonomy in the design. We will try to find an answer to the question: *Does your design change much with the support and help of taxonomy or not?*

Second, evaluate the usability of the user interface of the Hypermedia Representation of the Taxonomy [Gabbard and Hix, 1997] and recommend alternatives to improve human computer interface of the application and iteratively improve this interface.

Method

After greetings, the purpose of the experiment explained to the participants (see Appendix A). They are informed that they are free to withdraw from experiment whenever they want. They are helping to evaluate the interface and the structure of the taxonomy. We are not evaluating them; instead we are dealing with the usability of the interface. If they do an error, it is not theirs, it is application's error. Their data will be used just for research purposes not for commercial purposes and no names will be presented in the data. We emphasized that they should think aloud in order to collect data.

Second, they signed a series of consent forms (see Appendix B) and filled in a pre-questionnaire (see Table 9). We thought that experience of the participants with VEs may play an important role in this experiment. So we try measure their levels with a simple pre-questionnaire.

PRE-QUESTIONNAIRE

1. How well do you know the VE devices such as 3D mice, HMDs, gloves etc.?

   a few (1) (2) (3) (4) (5) (6) (7) a lot

2. Have you ever participated in any VE application?

   Yes No

3. Have you ever designed a VE application?

   Yes No

   If YES, please answer 4

4. Did you considered it as a user-centered (user friendly) VE application?

   Yes No

Table 9. Pre-Questionnaire

Third, they read the proposed VE design scenario (see Table 8). We left them free to think over the scenario for a few minutes. They are told that they can *use pencil and paper* and/or can *tell us* about their design whichever way they prefer. They studied the scenario either on a paper or directly told us what they think. They used paper to take notes or to arrange their thoughts. When they were silent, we encouraged them to think aloud. We waited and took notes until they said that their design is finished.

And then, we showed web version of the taxonomy and wanted them to reconsider their design. We looked for how the taxonomy is affecting their decisions. Meanwhile we encouraged them to talk about the interface. What do they liked or disliked? Is it helpful or not? We tried to collect *subjective* and *qualitative* data. Therefore, used following qualitative data generating techniques:

- Concurrent verbal protocol taking (thinking aloud)
- Critical incident taking
- Structured interviews

During experiment, we observed the behaviours of the participants and took notes. We also noted their hot comments about design and interface.

We used r*eal-time note-taking* as data collection technique.

Equipment

The experiment conducted using a personnel computer in MOVES Lab at NPS. The web site was installed in a local computer and that machine was used during the whole experiment.

Risks

This research involves no risks or discomforts greater than those encountered in daily life.

Safety Measures

The evaluator presented continuously and monitored the safety of the procedure.

Participants

Nine volunteers participated in almost 45 minutes session.

Confidentiality

Collected data will not be associated with the name of the participants. Each participant received a random number, which served to identify participant with results and questionnaires.

Consent

Participants asked to sign a series of consent forms (Appendix B) before the start of the experiment. Participants were given the names and telephone numbers of the evaluator so that they could be able to voice any concerns at any time.

4. Pilot Testing

Finally, all the settings and procedures have been determined and we did a pilot testing to ensure that all parts of the experiment are ready. We did not want the hardware or software to crash during an experimental session.

The experimental tasks (in our case *scenario*) should be completely run through at least once, using the intended hardware and software (i.e., the interface prototype) by someone other than the person(s) who developed the tasks, to make sure, for example, that the prototype supports all the necessary user actions and that the instructions are unambiguously worded [Hix and Hartson, 1993]. Like so, we wanted to minimize the possibilities for problems that might invalidate a test session.

We just used a volunteer to test our hardware, software, experimental procedures and instructions. At the beginning of the experiment we thought that one session is going to last approximately 30 minutes. During pilot testing this time went up to 45 minutes and we corrected experiment time. We caught some important points for evaluator to be cautious.

First part of the experiment was tend to be time consuming and then little amount of time left for second phase which is much more important for us. The evaluator has to be careful to regulate the time between two phases. A reminder: First phase is design of scenario without web site while second phase is redesign of scenario with web site help and support.

The evaluator has to be cautious to get ideas of the participants without helping them to do tasks in the scenario. Sometimes participants may think silent

and that does not help much to us. In this case, be careful and prod the participants by not causing them to feel that they are being prodded.

D. COLLECTING THE DATA

Subjective and *qualitative* data with nine participants collected. We used following qualitative data generating techniques:

- Concurrent verbal protocol taking (thinking aloud)
- Critical incident taking
- Structured interviews

Real-time note-taking was the data collecting technique.

The participants sat in front of a computer and evaluation session started like so. The comments of the participants and observations of the evaluator recorded during the evaluation. Pen and paper used for recording tools. When we took notes, participants saw that we were recording their comments.

E. DIRECTING THE EVALUATION SESSION

We try not to affect the participants' thoughts during the session. Most of them gave good feedbacks about the interface and usage of the taxonomy without prodding to get their thoughts. They also participated in prior experiments in NPS, because of this; they showed no enthusiasm or fear. They were open-minded and stated their thoughts very clearly. A couple of them studied on a paper silently at the beginning, but we prod them get their thoughts and observations.

F. ANALYZING THE DATA

Data is recorded for each participant separately and organized later. We merged all data and presented them as whole. Because some comments, thoughts and recommendations became the same after a while. The organized data will be presented in the next Chapter – Usability Evaluation Results.

This data was analyzed and some recommendations included in the current version of the application.

G. DRAWING CONCLUSIONS TO FORM A RESOLUTION FOR EACH PROBLEM

The problematic areas determined and then tried to find a resolution for each of them. Detailed information will be presented in the next Chapter – Usability Evaluation Results.

H. REDESIGNING AND IMPLEMENTING THE REVISED INTERFACE

After determination of the problems related to user interface, possible recommendations applied to the current user interface. Thus, a much more effective and user friendly interface implemented for the application.

THIS PAGE INTENTIONALLY LEFT BLANK

V. USABILITY EVALUATION RESULTS

A. OVERVIEW

Data will be presented in three parts in the next section. First part will be the general comments, problems and recommendations about overall web site. Second will be more detailed and will consist of page by page presentation. The usage of the taxonomy will be the third part. After this, data will be analyzed in the same structure. We went over every suggestion and stated our thoughts. In the last sub-section we presented redesigned web site.

B. COLLECTED DATA

1. Overall Data about Web Site

Data about overall web site presented as follows in Table 10.

No	Comments
1	Footer links are absent. If it can be added, the efficiency of site may increase.
2	User may need for .pdf or .ppt files if available.
3	Overview and Descriptions pages design are not consistent. Go to the Top links are absent in the overview page.
4	There is no link to web master.
5	There may be some links to the other VEs sites.
6	*Labels* are meaningless for some participants.
7	*Taxonomy* is confusing maybe, more clear word needed like *Design of VEs…*
8	Additional media types may make the web site more powerful and better.
9	Font size of sub-titles in the *descriptions* and *overview* pages may be smaller.
10	An advanced version may be according to the screen resolution.
11	More figures, graphics…

Table 10. Overall Web Site Data

2. Page by Page Data

a. Home

Look at Figure 31 for Home page information. The collected data presented in Table 11.

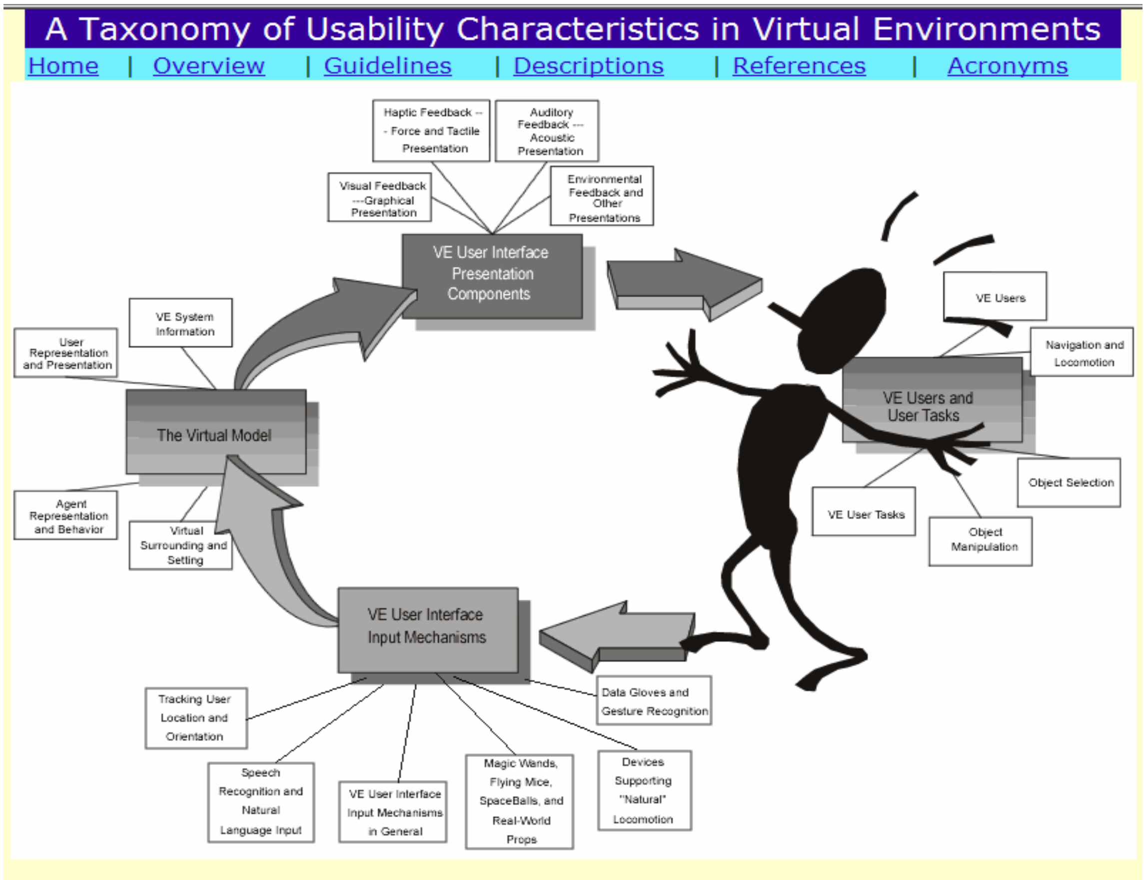


Figure 31. Home Page

No	Comments
1	Home page (overall figure) fonts are too small and not readable. When the mouse is over the text boxes, they may get big enough to read. The links on the text boxes are not recognizable very easily. Mouse turns to a hand shape to show the link.
2	Home page does not give information about the purpose of the web site. A short explanation like *abstract as in papers* may be more helpful.(Note: Not all of them stated this)
3	Black and white page, it is not good for a web site application.

4	Figure flow is good and intuitive after examining a couple seconds.
5	VE not represented in overview figure, display some stuff that represents VEs like computer picture that represents computer related things.
6	In overall picture *presentations components* are confusing and too long. *Users, Input, Model* and *Output* may be used for main areas. It is much clearer.
7	There is a misunderstanding in overall picture. There are four main areas. When you click the main area box it takes you the first table of that area. User has an expectation that when he clicked that link he supposed to find a summary table about that area.

Table 11. Home Page Data

b. Overview Page

A portion of this page is presented in Figure 32 to give an idea. Collected data related to this page presented in Table 12.

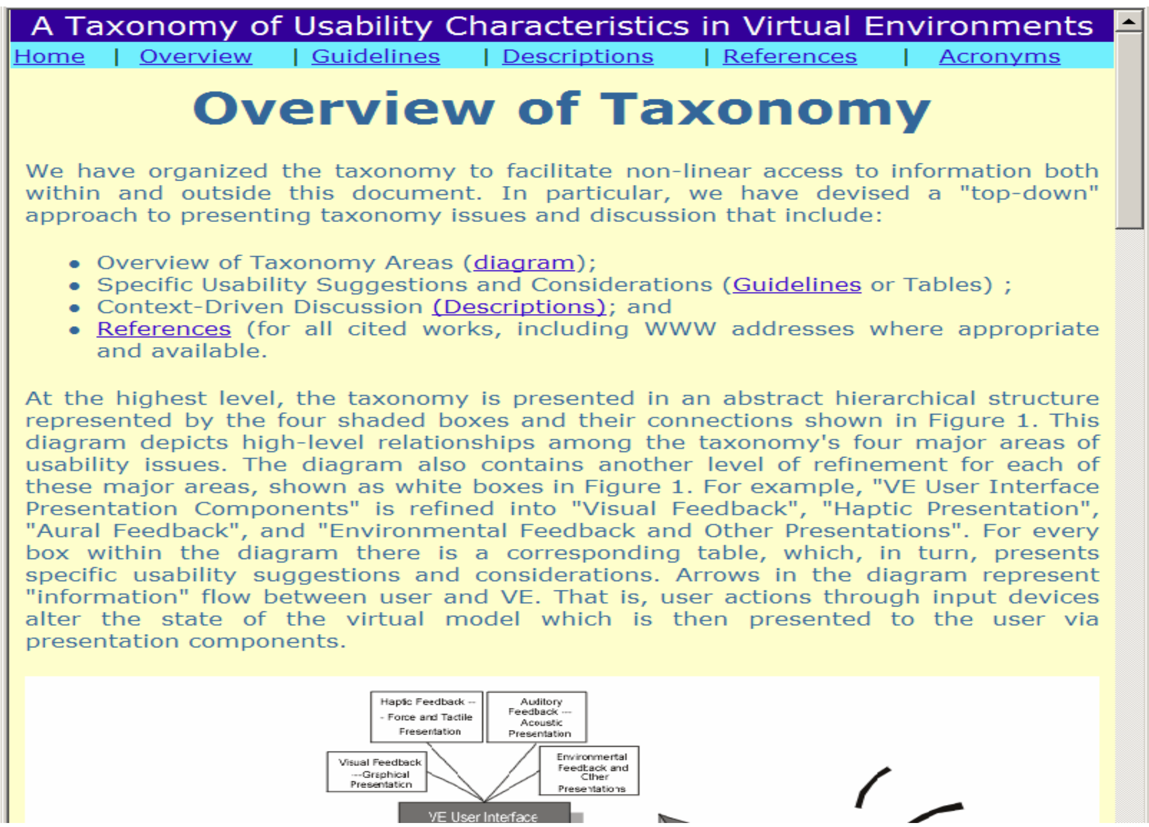


Figure 32. A Portion of Overview Page

No	Comments
1	Sub-sections may be on the left column and this will be more helpful for navigation purposes.
2	Inconsistent with *Descriptions* page. *Go to Top of the document* link is absent.

Table 12. Overview Page Data

c. Guidelines Page

Look at Figure 33 for Guidelines page design. Collected data presented in Table 13.

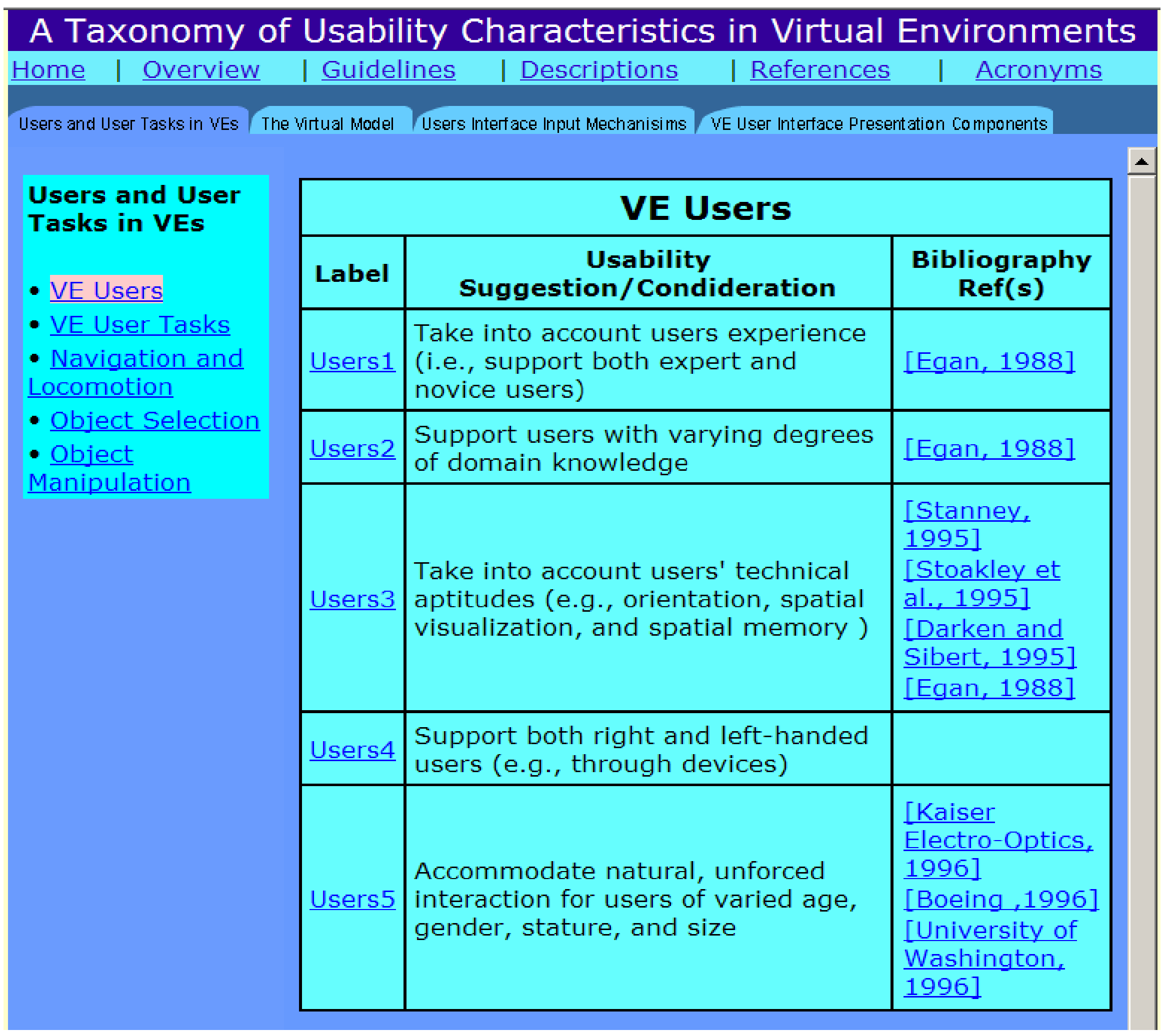


Figure 33. Guidelines Page

No	Comments
1	Navigation design is really good.
2	Search for *specific topic* in the guidelines may be more helpful. E.g. I want to see the guidelines about …..
3	The background (blue) is flashing—too bright.
4	A link in the *guidelines* table that directly takes you to the beginning of related *descriptions* page where the table content is discussed may be helpful.
5	*Labels* are not clear. Instead of labels, a short description of that guideline may be used. It will decrease the *understanding* and *searching* time.
6	In guidelines table, *labels* may be non-sense for users. Put the link to the guidelines and remove the *labels*.

Table 13. Guidelines Page Data

d. Descriptions Page

Look at Figure 34 for Descriptions page information. Collected data presented in Table 14.

No	Comments
1	Background color is good.
2	There are some blue italic fonts that are the same color with link and that is confusing.
3	Pages are too long vertically —too much scrolling
4	The descriptions are too long, I am lost. A small picture may be helpful to show where I am.
5	Acronyms in the title are not good.
6	Guidelines are emphasized with italic-bold fonts which is very good.
7	In context explanations, most important things must be discussed before and explain the details later.

8	*References* can be linked to their sources, if possible, and that would be more helpful.
9	Sub-sections may stay on the left column and this will be more helpful for navigation.
10	Picture quality is poor. Resolution is bad.(e.g. CAVE picture)
11	Instead of pictures graphs may be more helpful. Graphs show the details much more clear like in CAVE picture. The details are lost.
12	Some figures are too small.

Table 14. Descriptions Page Data

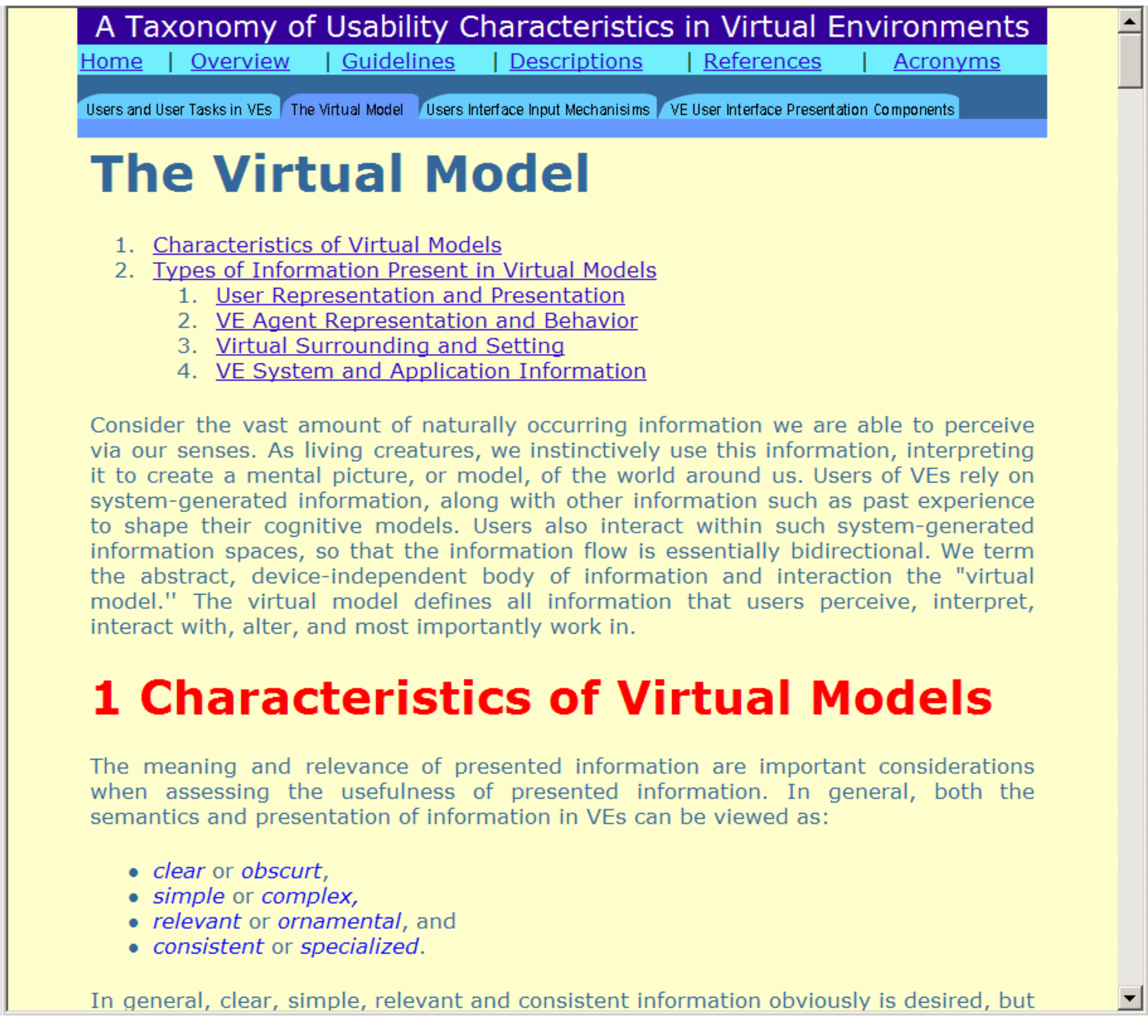


Figure 34. A Portion of Descriptions Page

e. References Page

Look at Figure 35 for References page design. Collected data presented in Table 15.

No	Comments
1	In *References* page *next*, *previous* … fonts are not recognizable, the font size may be bigger.
2	A search engine in the references page would be more helpful. The explanations for references must be more detailed. At least abstract length information must be placed.
3	While navigating the references, table height does not stay fix which distract the attentions.
4	Reference number in the table is unnecessary. References are already sorted alphabetically.
5	For References, using selectable number of records at a time may be more helpful like 5, 10, 20… record at a time.

Table 15. References Page Data

f. Acronyms Page

Look at Figure 36 for Acronyms page design. Collected data presented in Table 16.

No	Comments
1	It is a good idea to use this page. Well designed.

Table 16. Acronyms Page Data

A Taxonomy of Usability Characteristics in Virtual Environments

Home | Overview | Guidelines | Descriptions | References | Acronyms

REFERENCE INFORMATION

No	Abbrivation	Explanation
7	[Barfield et al., 1995]	Barfield, W., Zeltzer, D., Sheridan, T., and Slater, M. (1995). Presence and performance within virtual environments. In Virtual Environments and Advanced Interface Design, chapter 12, pages 473-513. Oxford University Press.
6	[Barfield et al., 1997]	Barfield, W., Hendrix, C., and Bystrom, K. (1997). Visualizing the structure of virtual objects using head tracked stereoscopic displays. In 1997 IEEE Virtual Reality Annual International Symposium Proceedings, pages 114-119.
9	[Benford et al.,1995]	Benford, S., Bowers, J., Fahlen, L. E., Greenhalgh, C., and Snowdon, D. (1995). User embodiment in collaborative virtual environments. In Human Factors in Computing Systems, CHI '95 Conference Proceedings, pages 242-249.
8	[Benford,1996]	Benford, S. (1996). Shared spaces: Transportation, artificiality, and spatiality. In Computer-Supported Cooperative Work (CSCW '96) Conference Proceedings, pages 77-86.
10	[Bennet et al., 1996]	Bennett, D., Chapelle, B. D. L., Zeltzer, D., Bryson, S. T., and Bolas, M. (1996). Information from the SIGGRAPH '96 Panel Session, "The Future of Virtual Reality: Head Mounted Displays versus Spatially Immersive Displays".

First Previous Next Last

Records 6 to 10 of 157

Figure 35. References Page

A Taxonomy of Usability Characteristics in Virtual Environments

Home | Overview | Guidelines | Descriptions | References | Acronyms

Acronyms

BOOM	Binocular Omni-Oriented Monitor
CAD	Computer-Aided Design
CAVE™	Cave Automatic Virtual Environment
CHI	Computer-Human Interaction
CSCW	Computer-Supported Cooperative Work
DIVE	Distributed Interactive Virtual Environments

Figure 36. A Portion of Acronyms Page

3. Taxonomy Usage

The Taxonomy usage way is differed according to the user knowledge/skill level about *characteristics of VE devices* and *previous knowledge about Taxonomy*. Also their experience in VE applications was very dominant factor on how to use the Taxonomy.

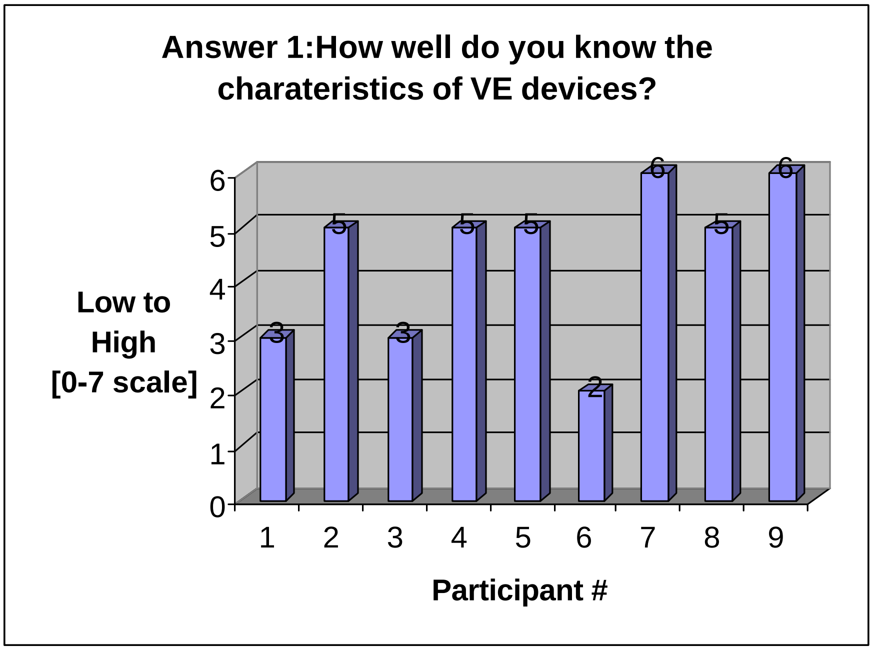


Figure 37. Pre-Questionnaire Result

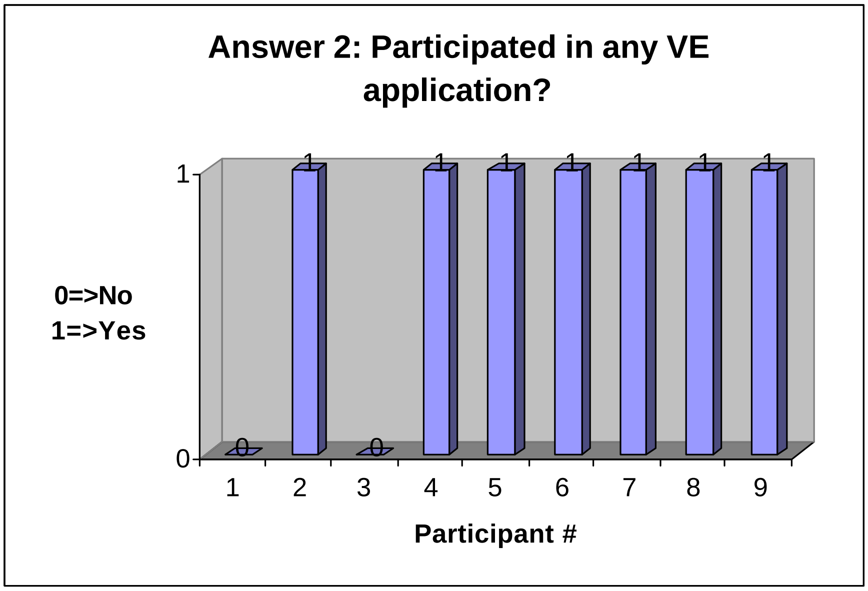


Figure 38. Pre-Questionnaire Result II

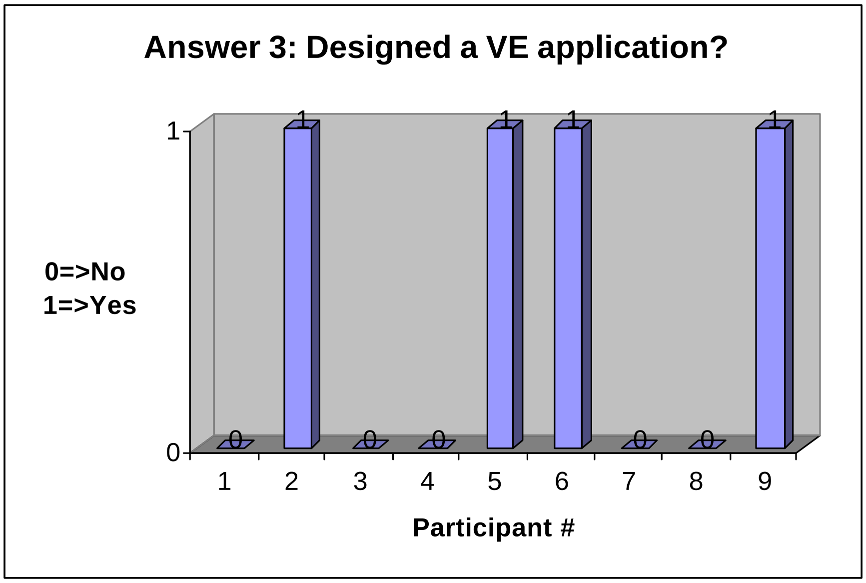


Figure 39. Pre-Questionnaire Result III

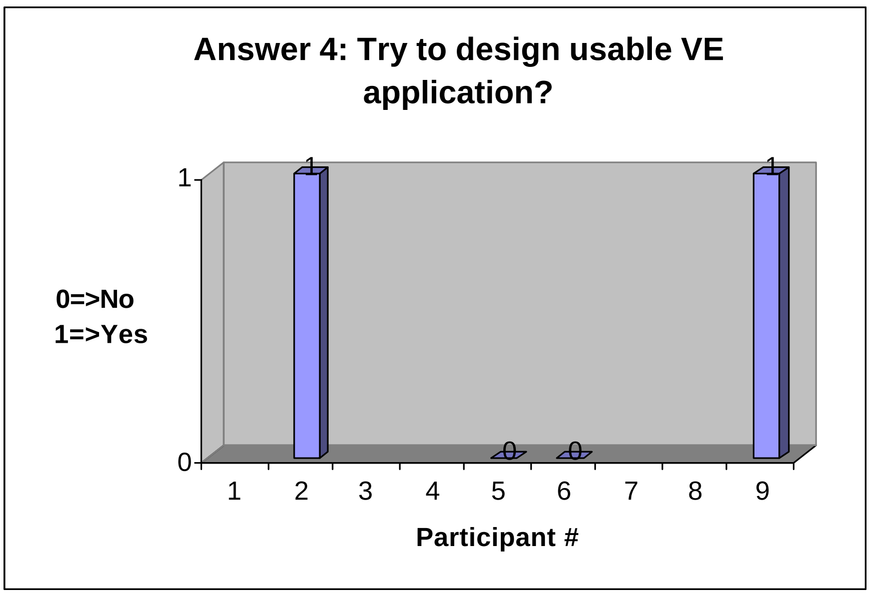


Figure 40. Pre-Questionnaire Result IV

When we look at the pre-questionnaire results, we saw that the level of participants' knowledge about VE devices is not so bad. On the other hand, they designed *very few* VEs or *never*. In their designs, usability was not an important factor. Their approach is that if it is usable then it's good; but if not, it still can be used (see Figures 37-40).

The participants who have low level knowledge about VE devices, most of the time, tend to read the descriptions all or they looked for the synopsis of guidelines/descriptions to get much knowledge in short amount of time for reading or searching purposes. They spend their time in reading the context-driven discussion pages.

Skilled participants usually looked at the overview picture and added the areas which they forget to consider in their designs. For example auditory feedback is forgotten by some participants. When they see the overall picture they reconsidered their design and improved their VE design. After that, they looked at some boxes (guideline table titles in overall figure) which they thought may be related to their design in detail. They seek for guidelines which may help to improve their design. If the guideline is not clear they look for the descriptions for detailed information. Very few participants felt the need for looking at the references for more information. They just looked the references to test the web site if it is working or to find what kind of information the references page/link offers.

After understanding the purpose of overall figure, participants find it very helpful for their design. But most of them couldn't improve their initial design. It was time consuming to use the taxonomy for the very first time and they had a limited time. Instead they looked some areas which interests them and developed these areas.

Another reason for not improving their design may be that this is just an experiment. They are not going to produce an application to sell and they have nothing to loose if their product is not good.

C. ANALYSIS OF THE DATA

1. Overall Data Analysis

Data about overall web site is analyzed as is Table 17.

No	Comments	Reconsideration/Resolution
1	Footer links are absent. If it can be added, the efficiency of site may increase.	That's a good idea. The footer link will be added. Implementation is easy and importance is medium.
2	User may need for .pdf or .ppt files if available.	We have .pdf of the whole document. Put that document in site.
3	Overview and Descriptions pages design are not consistent. *Go to the Top of the document* links are absent in the overview page.	Add *Go to the Top of the document* links to *Overview* page.
4	There is no link to web master.	Put a link to webmaster inside the footer.
5	There may be some links to the other VEs sites.	It's very easy to add but importance is very low. One participant felt that need.
6	*Labels* are meaningless for some participants.	For advanced users, *labels* are necessary and give feedback to users when he clicked form guideline to descriptions if he is at correct place.
7	*Taxonomy* is confusing maybe, more clear word needed like *Design of VEs…*	Taxonomy is more comprehensive than proposed solution.
8	Additional media types may make the web site more powerful and better.	Revised version of this site may add these. That may really be beneficial.
9	Font size of sub-titles in the *descriptions* and *overview* pages may be more small.	Reformat the sizes of the headers.

10	An advanced version may be according to the screen resolution.	That's a good idea.
11	More figures, graphics…	Revised version of this site may add these. That may really be beneficial.

Table 17. Overall Web Site Data Analysis

2. Page by Page Data Analysis

a. Home

Data about Home page is analyzed as in Table 18.

No	Comments	Reconsideration/Resolution
1	Home page (overall figure) fonts are too small and not readable. When the mouse is over the text boxes, they may get big enough to read. The links on the text boxes are not recognizable very easily. Mouse turns to a hand shape to show the link.	That's a good idea. Implement like proposed add different colors to four main areas.
2	Home page does not give information about the purpose of the web site. A short explanation like *abstract as in papers* may be more helpful.(Note: Not all the participants stated this)	Add a short explanation which tells about the purpose of the site.
3	Black and white page, it is not good for a web site application.	Figure will be colored.
4	Figure flow is good and intuitive after examine a couple seconds.	GOOD.
5	VE not represented in overview figure, display some stuff that represents VEs like computer picture that represents computer related things.	If we add extra pictures inside the figure, it may seem messy. Keep it as simple as possible.
6	In overall picture *presentations components* are confusing and too long. *Users, Input, Model* and *Output* may be used for main areas. It is much clearer.	We left this decision to the authors of taxonomy. It is valid for novice users, on the other hand, experienced users may chose the original explanations.

7	There is a misunderstanding in overall picture. There are four main areas. When you click the main area box, it takes you the first table of that area. User has an expectation that when he clicked that link he supposed to find a summary table about that area.	One main box takes you to the summary guideline table while the others to the first table of that area. In the future, a summary table may be added to the other three areas. One participant recognized this and others did not see this confusing.

Table 18. Home Page Data Analysis

b. Overview Page

Data about Overview page analyzed as in Table 19.

No	Comments	Reconsideration/Resolution
1	Sub-sections may be on the left column and this will be more helpful for navigation purposes.	This may take the screen space and left a narrow space for context section. As a result we can see imbalanced screen.
2	Inconsistent with *Descriptions* page. *Go to Top of the document* link is absent.	Correct inconsistencies.

Table 19. Overview Page Data Analysis

c. Guidelines Page

Data about Guidelines page analyzed as in Table 20.

No	Comments	Reconsideration/Resolution
1	Navigation design is really good.	GOOD.
2	Search for *specific topic* in the guidelines may be more helpful. E.g. I want to see the guidelines about …..	Put a search engine. Importance high and cost is 1.5 hour work.
3	The background (blue) is flashing — too bright.	Use a pastel color for background.
4	A link in the *guidelines* table that directly takes you to the beginning of related *descriptions* page where the table content is discussed may be helpful.	That is not so important. One participant needed this.

5	*Labels* are not clear. Instead of labels, a short description of that guideline may be used. It will decrease the *understanding* and *searching* time.	At first, labels may be meaningless for novice users. Even though they seem meaningless, they are still giving feedback to users when navigating between guidelines and descriptions pages. You can see the labels and say that I am at the correct section/part of the page. In the long run, experienced users may need them.
6	In guidelines table, *labels* may be non-sense for users. Put the link to the guidelines and remove the *labels*.	

Table 20. Guidelines Page Data Analysis

d. Descriptions Page

Data about Descriptions page is analyzed as in Table 21.

No	Comments	Reconsideration/Resolution
1	Background color is good.	GOOD.
2	There are some blue italic fonts that are the same color with link and that is confusing.	Change the emphasized or italic blue colored fonts to another color.
3	Pages are too long vertically — too much scrolling	It is very important for users to know where they are. And also most of the users hate from scrolling too. We are going to put a small version of overview figure to *show where you are*, to *minimize memory load* to hold the mental model of the system, and *to navigate* with help of this figure.
4	The descriptions are too long, I am lost. A small picture may be helpful to show where I am.	
5	Acronyms in the title are not good.	It's a good idea not to use acronyms in titles but it is not so important. One user suggested this.
6	Guidelines are emphasized with italic-bold fonts which is very good.	GOOD.

7	In context explanations, most important things must be discussed before and explain the details later.(Bottom line-up front)	This approach may be used in future design. Now, we are using the current document.
8	*References* can be linked their sources, if possible, and that would be more helpful.	It is very hard to update the hyper link information. Taxonomy has more than 150 sources. They are very akin to change. You can find on-line sources with any search engine in the www very easily.
9	Sub-sections may stay on the left column and this will be more helpful for navigation.	This may take the screen space and left a narrow space for context section. As a result we can see imbalanced screen.
10	Picture quality is poor. Resolution is bad.(e.g. CAVE picture)	We tried to use the best picture we have.
11	Instead of pictures, graphs may be more helpful. Graphs show the details much more clear like in CAVE picture. The details are lost.	This may be considered in future version.
12	Some figures are too small.	If we can find good resolution pictures, we can change and resize these figures or pictures.

Table 21. Descriptions Page Data Analysis

e. References Page

Data about References page is analyzed as in Table 22.

No	Comments	Reconsideration/Resolution
1	In *References* page *next, previous …* fonts are not recognizable, the font size may be bigger.	That is a good catch. Use different font size and color to make it distinguishable.
2	A search engine in the references page would be more helpful. The explanations for references must be more detailed. At least abstract length information must be placed.	Search engine is a good idea. It's importance high and coast is 1 hour.

3	While navigating the references, table height does not stay fix which distract the attentions.	Try to make the table height fix.
4	Reference number in the table is unnecessary. References are already sorted alphabetically.	Remove the reference number.
5	For References, using selectable number of records at a time may be more helpful like 5, 10, 20… record at a time.	Good idea for future version.

Table 22. References Page Data Analysis

f. Acronyms Page

Data analysis about Acronyms page is presented in Table 23.

No	Comments	Reconsideration/Resolution
1	It is a good idea to use this page. Well designed.	GOOD.

Table 23. Acronyms Page Data Analysis

D. REDESIGN

After analyzing the data as seen in previous section, we try to add the features that we see helpful to improve the interface.

We added the footer to the whole site. Footer links contains the navigation bar, link to web master, link to .pdf form of the taxonomy and copy right explanations (see Figure 41).

After that we made global changes to the site. First we started with Cascading Style Sheets (CSSs) and templates that used inside the site. Header font sizes rearranged and italic font color changed to a different color other than link color which is blue (see this at the bottom of Figure 41). Footer added to templates. Likewise we try to be consistent as much as possible.

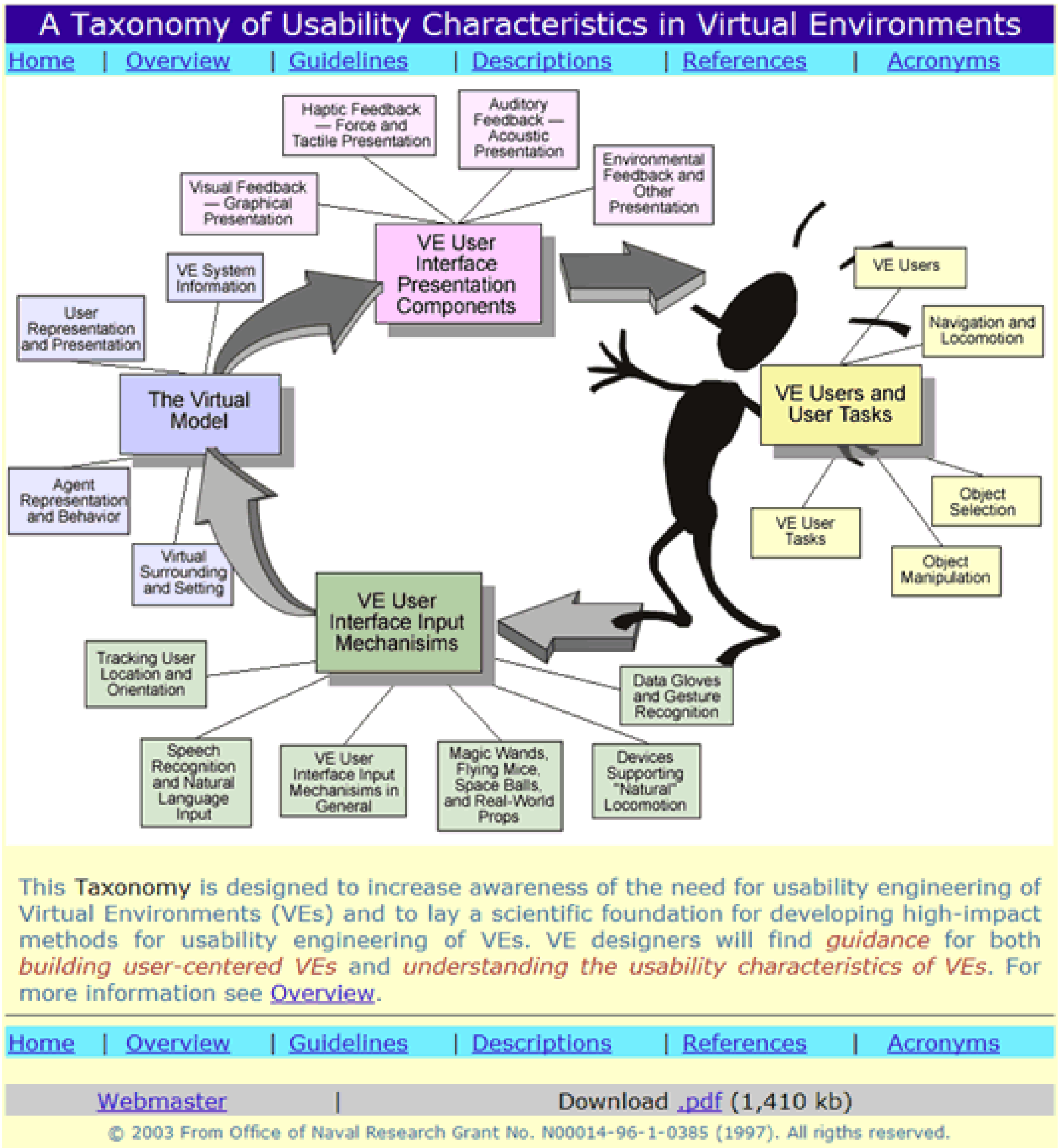


Figure 41. Redesigned Home Page

After that we started to modify the site page by page. Our first stop was Home Page. We redesigned the overview figure. In our design we tried to bring forward the dynamic property of the figure. To do that, we used different colors for four main areas (see Figure 41). The text boxes font sizes are made bigger and readable. In order to show the dynamic character of the figure, we used

swap image property which was swapping the text box with a bigger text box that is filled a little bit darker and different font color (see Figure 42).

A portion of the overview figure to show the dynamic behaviour of it. When the mouse is over the text boxes, text boxes immediately get bigger and show that they have dynamic property. The font size gets bigger and changes color. The color that fills in the text box also gets a little bit darker.

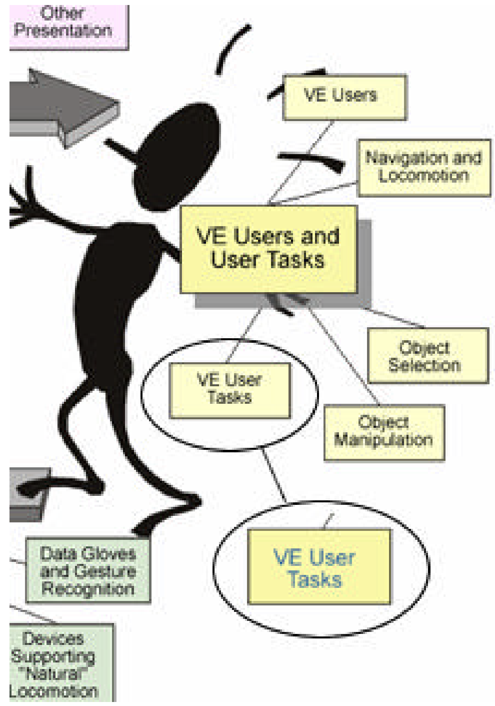


Figure 42. Dynamic Behavior of Overview Figure

A short explanation about the purpose of the site added near the bottom of Home Page (see Figure 41).

Overview Page made consistent with Descriptions Page by adding *go to top of the document* links.

The Guidelines Page redesigned by adding new features (see Figure 43). Background color changed to a pastel color. A search engine added to search in guidelines. You can search in the guidelines and references fields.

After making global changes to the site we just added a small version of the overview figure to the Descriptions Page. This figure is used to *show where you are*, to *minimize memory load* to hold the mental model of the system, and *to easily navigate* with help of it.

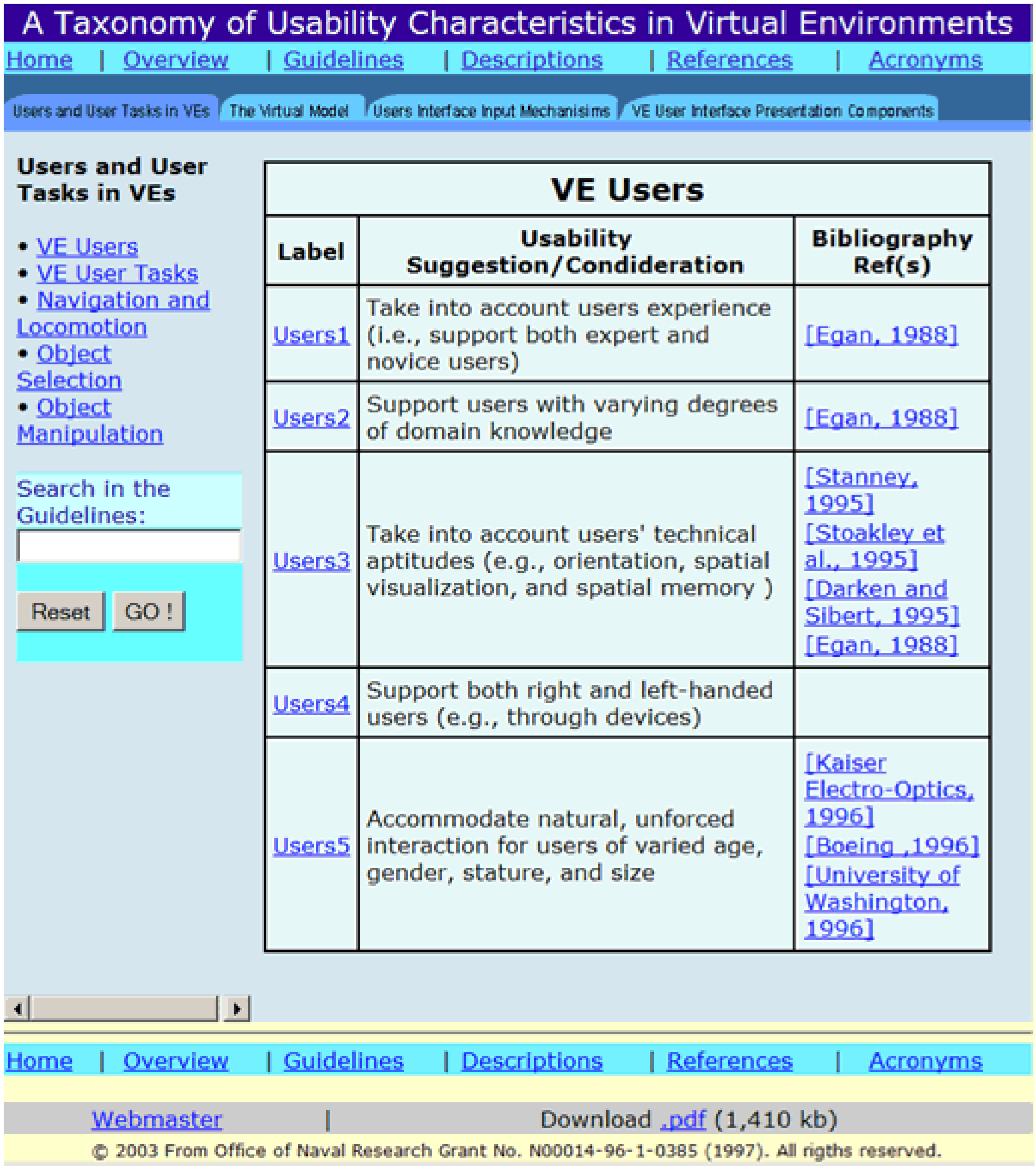
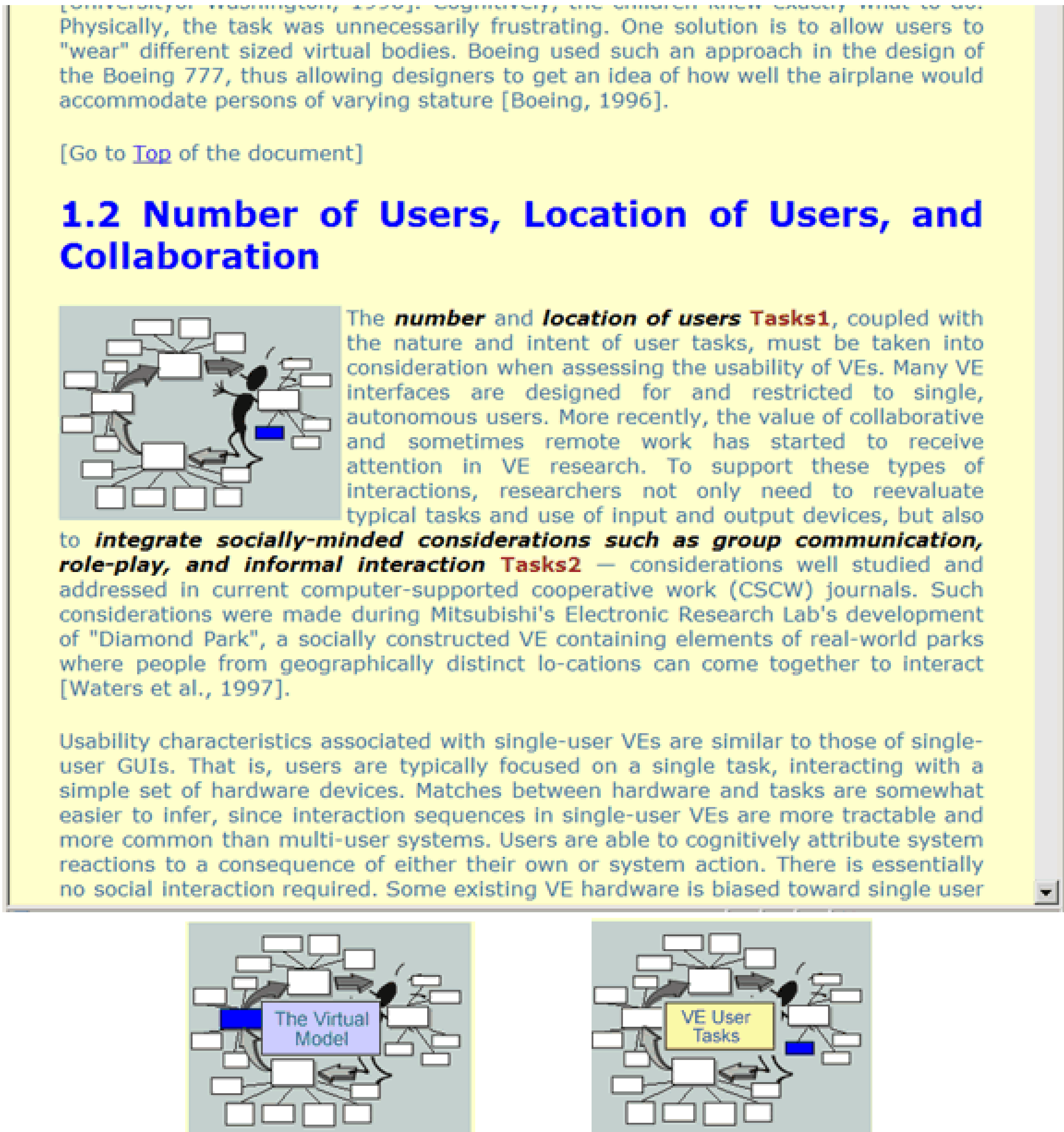


Figure 43. Redesigned Guidelines Page

As you can guess, small overview figure in Descriptions page (see Figure 44) also has dynamic behaviors. When you roll the mouse over the small rectangles inside the small figure, text box pops up in the middle of figure which says the name of that text box. When you clicked that rectangle, this takes you the place where that context is discussed. You will see that one rectangle is filled

with blue which states that you are here. We saw before that each box represents a different guidelines table. When you clicked that text box, it takes you to the related guideline table. In this version of small overview figure, it takes you the context-driven section where that title is discussed.



Physically, the task was unnecessarily frustrating. One solution is to allow users to "wear" different sized virtual bodies. Boeing used such an approach in the design of the Boeing 777, thus allowing designers to get an idea of how well the airplane would accommodate persons of varying stature [Boeing, 1996].

[Go to Top of the document]

1.2 Number of Users, Location of Users, and Collaboration

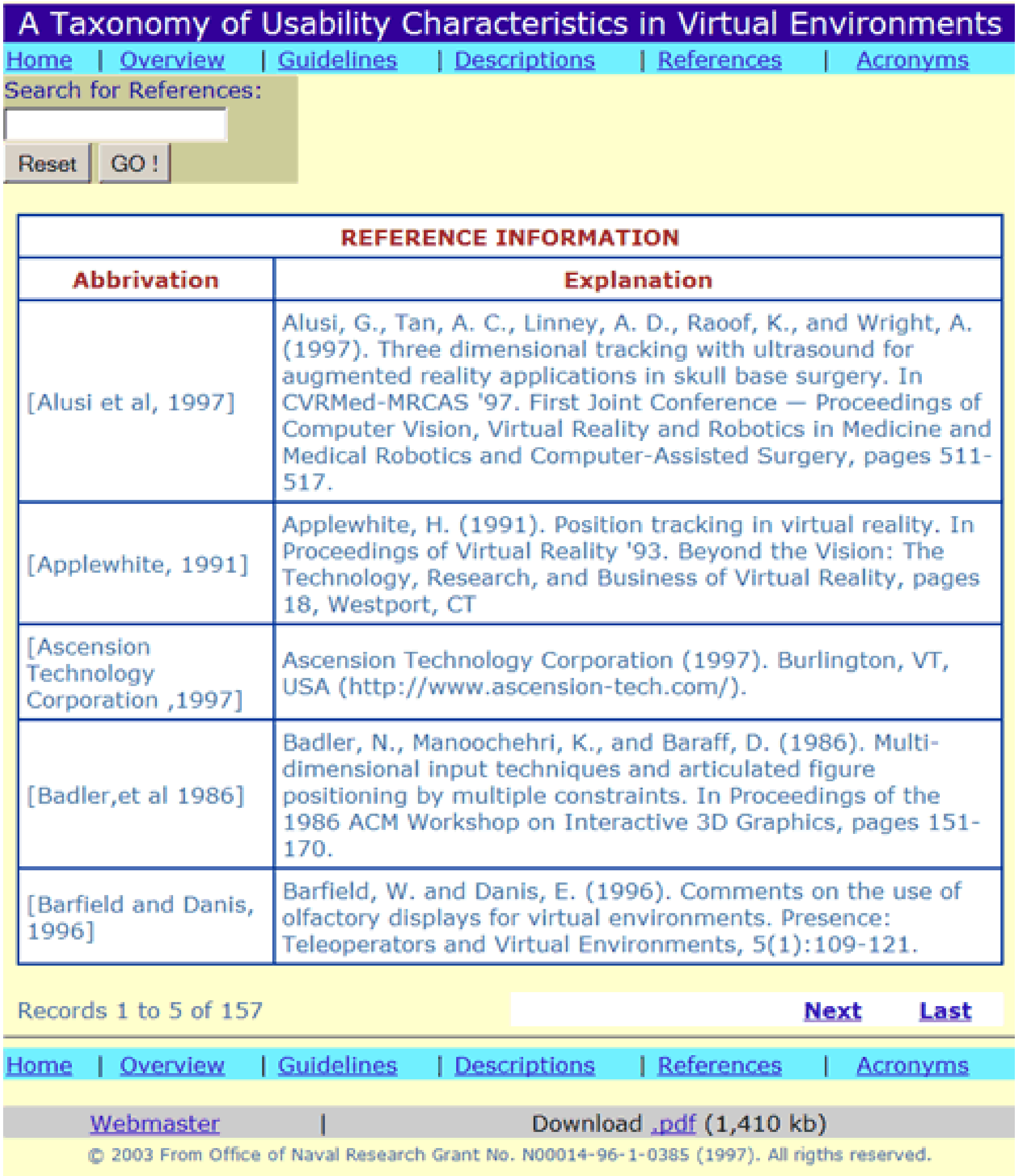
The ***number*** and ***location of users*** **Tasks1**, coupled with the nature and intent of user tasks, must be taken into consideration when assessing the usability of VEs. Many VE interfaces are designed for and restricted to single, autonomous users. More recently, the value of collaborative and sometimes remote work has started to receive attention in VE research. To support these types of interactions, researchers not only need to reevaluate typical tasks and use of input and output devices, but also to ***integrate socially-minded considerations such as group communication, role-play, and informal interaction*** **Tasks2** — considerations well studied and addressed in current computer-supported cooperative work (CSCW) journals. Such considerations were made during Mitsubishi's Electronic Research Lab's development of "Diamond Park", a socially constructed VE containing elements of real-world parks where people from geographically distinct lo-cations can come together to interact [Waters et al., 1997].

Usability characteristics associated with single-user VEs are similar to those of single-user GUIs. That is, users are typically focused on a single task, interacting with a simple set of hardware devices. Matches between hardware and tasks are somewhat easier to infer, since interaction sequences in single-user VEs are more tractable and more common than multi-user systems. Users are able to cognitively attribute system reactions to a consequence of either their own or system action. There is essentially no social interaction required. Some existing VE hardware is biased toward single user

Figure 44. A Portion of Redesigned Descriptions Page

When you look at the bottom of Figure 44, you will see two samples of small overview figure. This picture shows you how small overview figure works. The mouse is rolled over different rectangles and as a result, we got the names

of these rectangles. When you clicked these boxes, you will go to that section of context-driven pages.


A Taxonomy of Usability Characteristics in Virtual Environments

Home | Overview | Guidelines | Descriptions | References | Acronyms

Search for References:

Reset GO !

REFERENCE INFORMATION	
Abbrivation	**Explanation**
[Alusi et al, 1997]	Alusi, G., Tan, A. C., Linney, A. D., Raoof, K., and Wright, A. (1997). Three dimensional tracking with ultrasound for augmented reality applications in skull base surgery. In CVRMed-MRCAS '97. First Joint Conference — Proceedings of Computer Vision, Virtual Reality and Robotics in Medicine and Medical Robotics and Computer-Assisted Surgery, pages 511-517.
[Applewhite, 1991]	Applewhite, H. (1991). Position tracking in virtual reality. In Proceedings of Virtual Reality '93. Beyond the Vision: The Technology, Research, and Business of Virtual Reality, pages 18, Westport, CT
[Ascension Technology Corporation ,1997]	Ascension Technology Corporation (1997). Burlington, VT, USA (http://www.ascension-tech.com/).
[Badler,et al 1986]	Badler, N., Manoochehri, K., and Baraff, D. (1986). Multi-dimensional input techniques and articulated figure positioning by multiple constraints. In Proceedings of the 1986 ACM Workshop on Interactive 3D Graphics, pages 151-170.
[Barfield and Danis, 1996]	Barfield, W. and Danis, E. (1996). Comments on the use of olfactory displays for virtual environments. Presence: Teleoperators and Virtual Environments, 5(1):109-121.

Records 1 to 5 of 157 Next Last

Home | Overview | Guidelines | Descriptions | References | Acronyms

Webmaster | Download .pdf (1,410 kb)

© 2003 From Office of Naval Research Grant No. N00014-96-1-0385 (1997). All rigths reserved.

Figure 45. Redesigned References Page

Next, References Page redesigned as seen in Figure 45. After making small changes to the page, a search engine added as seen on left-upper corner of the page. A sample search for *Darken* is seen in Figure 46.

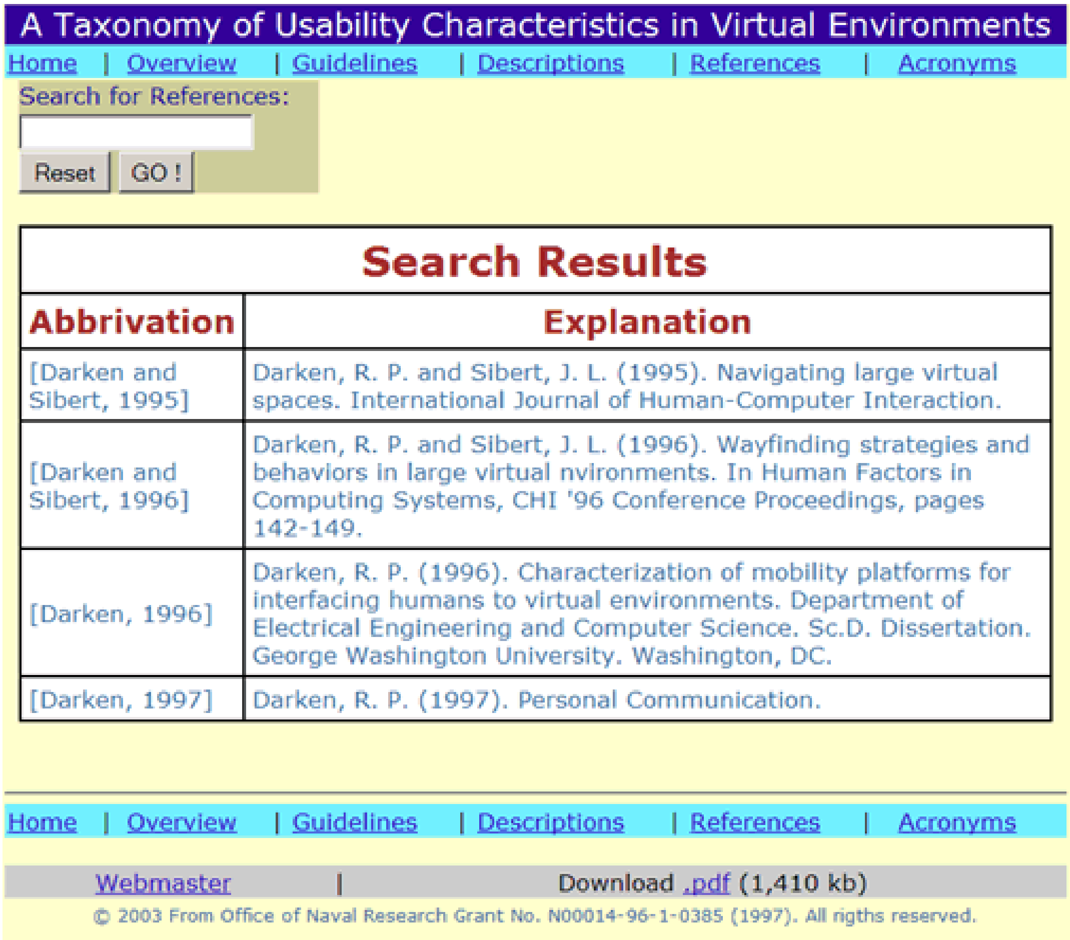


Figure 46. A Sample Search Result for *Darken* in References Page

THIS PAGE INTENTIONALLY LEFT BLANK

VI. CONCLUSIONS AND FUTURE WORK

A. CONCLUSIONS

We have developed the full WWW implementation of the taxonomy by using scenario based iterative formative usability evaluation. The non-linear nature of hypermedia is well-suited for the taxonomy. We think that we exploit the use of hyperlinks to provide a more *usable* and *navigable* document.

After implementation of WWW version of taxonomy, we are expecting researchers and developers to access the taxonomy very easily, as a result, they will have more growing tendency to use it. Therefore, taxonomy will help more users.

Web-based implementation is expected to be more beneficial because web will provide a widespread availability. Once available, we expect interested parties to *use* the taxonomy and provide feedback to aid in the constant process of updating and refining the taxonomy. We don't claim that we developed a perfect site. This site will get better as soon as the feedbacks and comments of users reach us. We will try to improve the interface and content of the site based on the user needs and comments.

This taxonomy will also serve as a foundation upon which development of new usability engineering methods for VEs can be based. Through iterative development, it may be used to refine a set of high-impact usability engineering methods specifically for VEs. Once developed, these methods in turn may be integrated into the overall system development lifecycle, creating better VEs which are less expensive to maintain, support, and use. The methods may also be used to evaluate existing VE applications, providing more *user oriented* requirements in subsequent releases [Gabbard and Hix, 1997]. From this point, Gabbard and others [1999] have developed a methodology that may benefit from this taxonomy.

While adding new studies to the taxonomy, we saw that it's not so easy to do so. There is no consistent set of rules for inserting new items to the taxonomy. People have to use their personnel judgments where to add their studies in the taxonomy. New studies have to be refined very carefully to prevent adding redundant things. You also must have a good knowledge about the *structure and context* of taxonomy and *usability characteristics* in VEs.

B. FUTURE WORK

We just converted text/paper form of the taxonomy to the web-based application as is and did not change the content of it. Taxonomy has written in 1997 and includes studies since that date. It's likely that there have been lots of researches and studies after 1997 about VEs. These are not included in the taxonomy therefore, the content update may be needed.

We did not provide direct links to specific VE products and applications mentioned in the taxonomy, and from cited literature to appropriate and available online papers and articles. We thought that the link addresses are changing very rapidly and they always need to be updated. It may need a special care and effort. On the other hand, implementation of these links is very easy.

Links to other resources also did not included, such as links to academic, commercial, and government VE research labs. A special separate page that covers these information and links may be added to the web site.

Individual taxonomy users may have different expectations from web-based taxonomy such as dynamic ordering and filtering based on their needs. For example, if an interested developer is researching usability issues of display devices, a re-ordered taxonomy could be generated which structures and ranks both explicit and implicit display issues. Although we put a simple search engine in the site, it may not meet user expectations. A more comprehensive and complex structure can be used to meet individual user needs after getting user expectations.

Another important issue is that administrator of this site may need an interface to edit the database. The limitations and design of this interface can be considered after what kind of changes is going to be made to the database by getting feedbacks from users.

While adding a new study to the taxonomy, we saw that it's not so easy to do so. Automating this process is an important issue and need some work. It would be nice if a researcher could submit a suggested update including the principle and references. This would go to a taxonomy administrator who would decide:

1. if it was good enough to include in the taxonomy, and
2. where it would go.

Then he would have to link it up and make it publicly available.

A future study may consider the points we emphasized above and then update taxonomy with web-site (re)design.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A: IN BRIEFING

Welcome to the Naval Postgraduate School Moves Department. My name is Asim TOKGOZ. Thank you for participating in this experiment. This experiment deals with the usability of a Taxonomy of Usability Characteristics in Virtual Environments.

This experiment does not test your intelligence or performance level in this type of an environment. Purpose is to try to find the best way to design user-centered virtual environments. Your performance will be used only for research purposes, and it will not be used in any type of records. Prior to starting the experiment you will be asked to read and sign a series of consent forms and then fill in a questionnaire. Please read them carefully and ask me if you have any questions. The experiment will take approximately 45 minutes. If you don't have any question, please read and sign the consent forms.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B: CONSENT FORMS

1. GENERAL

The forms in the appendix appear in the same format utilized for the experiment and do not follow the standard thesis formats utilized in the chapters of this document. This appendix consists of three documents: Consent Form, Minimal Risk Consent Statement, and the Privacy Act Statement. Each participant is required to read and sign these documents before he is allowed to participate in the study.

2. CONSENT FORM

PARTICIPANT CONSENT FORM

1. **Introduction.** You are invited to participate in a usability analysis study of a Taxonomy of Usability Characteristics in Virtual Environments. This research is aimed at measuring the help/guidance of the Taxonomy when designing the Virtual Environments. You will be given a VE scenario and construct your model according to that scenario. After that you will be allowed to look at the web version of taxonomy and you will be wanted to redesign the scenario. In redesign cycle it is very important to think aloud in order to collect the data concerning the experiment. Most of the data will be qualitative so I want to emphasis again that the thinking aloud is very important.

2. **Background Information.** Data is being collected by the Naval Postgraduate School's MOVES Department for use to develop user-centered virtual environments.

3. **Procedures.** If you agree to participate in this study, the researcher will explain the procedures in detail.

    - You will read the scenario
    - After that you will design the scenario by writing on a paper
    - Upon completion of paper prototype you will be introduced with web version of the Taxonomy
    - You will redesign the scenario with the help of Taxonomy

The total amount of time is approximately 45 minutes.

4. **Risks and Benefits.** The research involves no risk or discomforts greater than those encountered in ordinary use of desktop computers. The benefits to the participants will be to contribute to current research in advancing navigation metaphors in virtual environments.

5. **Compensation.** No tangible reward will be given. A copy of the results will be available to you at the conclusion of the experiment.

6. **Confidentiality.** The records of this study will be kept confidential. No information will be publicly accessible which could identify you as a participant.

7. **Voluntary Nature of the Study.** If you agree to participate, you are free to withdraw from the study at any time without prejudice. You will be provided a copy of this form for your records.

8. **Points of Contact.** If you have any further questions or comments after the completion of the study, you may contact the research supervisor, Dr. Rudolph P. Darken (831) 656 7588 darken@nps.navy.mil.

9. **Statement of Consent.** I have read the above information. I have asked all questions and have had my questions answered. I agree to participate in this study.

-- ------------------------
Participant's Signature Date

-- --------------------------
Researcher's Signature Date

3. MINIMAL RISK CONSENT STATEMENT

NAVAL POSTGRADUATE SCHOOL, MONTEREY, CA 93943
MINIMAL RISK CONSENT STATEMENT

Participant: VOLUNTARY CONSENT TO BE A RESEARCH PARTICIPANT IN:

The Usability Analysis of a Taxonomy Of Usability Characteristics In Virtual Environments

1. I have read, understand and been provided Information for Participants that provides the details of the below acknowledgments.

2. I understand that this project involves research. An explanation of the purposes of the research, a description of procedures to be used, identification of experimental procedures, and the extended duration of my participation have been provided to me.

3. I understand that this project does not involve more than minimal risk. I have been informed of any reasonably foreseeable risks or discomforts to me.

4. I have been informed of any benefits to me or to others that may reasonably be expected from the research.

5. I have signed a statement describing the extent to which confidentiality of records identifying me will be maintained.

6. I have been informed of any compensation and/or medical treatments available if injury occurs and is so, what they consist of, or where further information may be obtained.

7. I understand that my participation in this project is voluntary; refusal to participate will involve no penalty or loss of benefits to which I am otherwise entitled. I also understand that I may discontinue participation at any time without penalty or loss of benefits to which I am otherwise entitled.

8. I understand that the individual to contact should I need answers to pertinent questions about the research is Professor Rudy Darken, Principal Investigator, and about my rights as a research participant or concerning a research related injury is the Modeling Virtual Environments and Simulation Chairman. A full and responsive discussion of the elements of this project and my consent has taken place.

Medical Monitor: Flight Surgeon, Naval Postgraduate School

________________________________ ___________________________
Signature of Principal Investigator Date Signature of Volunteer Date

____
Signature of Witness Date

4. PRIVACY ACT STATEMENT

NAVAL POSTGRADUATE SCHOOL, MONTEREY, CA 93943
PRIVACY ACT STATEMENT

1. Authority: Naval Instruction

2. Purpose: THE USABILITY ANALYSIS OF A TAXONOMY OF USABILITY CHARACTERISTICS IN VIRTUAL ENVIRONMENTS.

3. Use: Physiological response data will be used for statistical analysis by the Departments of the Navy and Defense, and other U.S. Government agencies, provided this use is compatible with the purpose for which the information was collected. The Naval Postgraduate School in accordance with the provisions of the Freedom of Information Act may grant use of the information to legitimate non-government agencies or individuals.

4. Disclosure/Confidentiality:

   a. I have been assured that my privacy will be safeguarded. I will be assigned a control or code number, which thereafter will be the only identifying entry on any of the research records. The Principal Investigator will maintain the cross-reference between name and control number. It will be decoded only when beneficial to me or if some circumstances, which are not apparent at this time, would make it clear that decoding would enhance the value of the research data. In all cases, the provisions of the Privacy Act Statement will be honored.

   b. I understand that a record of the information contained in this Consent Statement or derived from the experiment described herein will be retained permanently at the Naval Postgraduate School or by higher authority. I voluntarily agree to its disclosure to agencies or individuals indicated in paragraph 3 and I have been informed that failure to agree to such disclosure may negate the purpose for which the experiment was conducted.

   c. I also understand that disclosure of the requested information is voluntary.

____

Signature of Volunteer   Name, Grade/Rank (if applicable)   Date

____

Signature of Witness   Date

LIST OF REFERENCES

Bricken, M. (1990). University of Washington Technical Report R-90-2, HITLab, *Virtual worlds: No Interface to Design*, 1990,. Technical report available online at http://www.hitl.washington.edu/publications/r-90-2/. Last accessed March 23, 2003

Carrol, J.M. and Rosson, M.B. (1985*) Usability Specifications as a Tool in Iterative Development*. In H.R. Hartson (Ed.), Advances in Human-Computer Interaction, Vol. 1, Pages 1-28, Norwood, NJ: Ablex, 1985.

Carrol, J.M., Singley, M.K. and Rosson, M.B.(1992*). Integrating Theory Development with Design Evaluation*. Journal of Behaviour and Information Technology, 1992.

Cockayne, W. and Darken, P.R. (in press). *The Application of Human Ability Requirements to Virtual Environment Interface Design and Evaluation*.

del Galdo, E.M., Williges, R.C., Williges, B.H., & Wixon, D.R. (1986). *An Evaluation of Critical Incidents for Software Documentation Design*. In Proceedings of Thirtieth Annual Human Factors Society Conference, Anaheim, CA., 1986.

Dick, W. and Carey, L. (1978). *The Systematic Design of Instruction*. Glenview, IL: Scott, Foresman 1978.

Desurvire H.W., Kondziela J.M., and Atwood M.E.(1992), *What Is Gained and Lost When Using Evaluation Methods Other than Empirical Testing*. People and Computers VII, Pages 89-102, Cambridge University Press, Cambridge, UK, 1992.

Gabbard, Joseph L. and Hix, Deborah (1997),Office of Naval Research Grant No. N00014-96-1-0385, *A Taxonomy of Usability Characteristics in Virtual environments*, November 1997.

Gabbard, Joseph L. and Hix, Deborah (1998), *Usability Engineering for Virtual Environments through a Framework of Usability Characteristics, 1998.*

Gabbard, J., & Hix, D. (2001). *Usability Engineering of Virtual Environments*. Virtual Environments Handbook, Kay M. Stanney, Ed., University of Central Florida, Chapter 39, 2001.

Gabbard, J., Hix, D., Swan II, E. (1999) *User-Centered Design and Evaluation of Virtual Environments*. IEEE Computer Graphics and Applications, Volume 19, pages 51-59, Number 6, November/December 1999.

Gabbard J.L, Swartz K., Richey K. and Hix D. (1999a), *Usability Evaluation Techniques: A Novel Method for Assessing the Usability of an Immersive Medical VE*, pages 165-170. Proc. Virtual Worlds and Simulation Conf. (VWSIM 99), Society for Computer Simulation Int'l, San Diego, CA, 1999.

Hackos, J.T and Redish, J.C (1998) *User and Task Analysis for Interface Design*, John Wiley and Sons, New York, 1998.

Hix, D. and Hartson, H.R. (1993). *Developing User Interfaces: Ensuring Usability through Product & Process*, John Wiley and Sons, Inc., 1993.

Hix D. , Swan E. II., Gabbard J.L., McGee, M., Durbin J. and King T. , (1999) *User-Centered Design and Evaluation of a Real-Time Battlefield Visualization Virtual Environment*, , pages 96-103, Proc. IEEE Virtual Reality 99 Conf., IEEE CS Press, Los Alamitos, CA, 1999.

Hannaford, B. and Venema, S. (1995). *Kinesthetic displays for remote and virtual environments.* In Virtual Environments and Advanced Interface Design, chapter 10, pages 415-436. Oxford University Press, 1995.

Hinckley, K., Pausch, R., Goble, J. C., and Kassell, N. F. (1994). *Design Hints for Spatial Input*. Pages 213-222. In Proceedings of ACM Symposium on User Interface Software & Technology (UIST '94). Document available online at

http://www.cs.virginia.edu/uigroup/publications/94/conferences/UIST/Survey.html

Mackey, W.E. and Davenport, G. (1989). *Virtual Video Editing in Interactive Multimedia Applications*, pages 802-810, Communications of the ACM, 32(7), 1989.

Nielson, J. and Molich, R. (1990*). Heuristic Evaluation of User Interfaces*. Pages 249-256. Proceedings of CHI Conference on Human Factors in Computing Systems, New York: ACM, 1990.

Nielson, J. (1992*). Finding Usability Problems through Heuristic Evaluation*. Pages 373-380. Proccedings of CHI Conference on Human Factors in Computing Systems, New York: ACM, 1992.

Nielsen J. (1993), *Usability Engineering*, Academic Press, Cambridge, MA, 1993.

Nielsen J. (1994), *Heuristic Evaluation, in Usability Inspection Methods*. Pages 25-62. John Wiley and Sons, New York, 1994.

Norman, D. A. and Draper, S. W. (1986). *Cognitive Engineering*, chapter 3. Lawerence Erlbaum Associates, 1986.

Rob, P. and Semann, E. (2000). *Databases: Design, Development, & Deployment Using Microsoft Access* McGraw-Hill Higher Education, 2000.

Salzman, M., Dede, C., and Loftin, R. B. (1995). *Usability and Learning in Educational Virtual Realities*. In Proceedings of the Human Factors and Ergonomics Society 39th Annual Meeting, pages 486-490, volume 1, 1995.

Siochi, A.C. and Ehrich, R.W. (1991). *Computer Analysis of User Interfaces Based on Repetition in Transcripts of User Sessions*. Pages 309-335. Transaction of Information Systems, 9(4), 1991.

Siochi, A.C. and Hix, D. (1991*). A Study of Computer-Supported User Interface Evaluation Using Maximal Repeating Pattern Analysis*. Pages 301-305.

Proceedings of CHI Conference on Human Factors in Computing Systems, New York: ACM, 1991.

Scriven, M. (1967). *The Methodology of Evaluation*. Pages 39-83. In R. Tyler, R. Gagne, and Scriven (Ed.), Perspectives of Curriculum Evaluation . Chicago: Rand McNally, 1967.

Siochi, A.C. and Ehrich, R.W. (1991). *Computer Analysis of User Interfaces Based on Repetition in Transcripts of User Sessions*. Pages 309-335. Transactions on Information Systems, 9(4), 1991.

Tokgoz, Asim (2002), *The Comprasion Of Spatial Knowledge Acquisition Between Egocentric and Exocentric Views In Non-Complex Virtual Environments*. Unpublished Paper as MV4002 course project, 2002.

Whiteside, J., Bennett, J., and Holtzblatt, K. (1988). *Usability Engineering: Our Experience and Evaluation*. Pages 791-817. In M. Helander(Ed.), Handbook of Human-Computer Interaction . Amsterdam: Elsevier North-Holland, 1998.

Wickens, C.D., (2002). *Situation Awareness and Workload in Aviation*, Blackwell Publishing Inc., pages 128-133, American Psychology Society 11(4), August 2002.

Williges, R.C. (1984). *Evaluating Human-Computer Software Interfaces*. Proceedings of International Conference on Occupational Ergonomics, 1984.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, VA

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, CA

3. Dr. Rudy DARKEN
   MOVES Institute
   Naval Postgraduate School
   Monterey,CA

4. CMDR Joe SULLIVAN
   MOVES Institute
   Naval Postgraduate School
   Monterey,CA

5. Asim TOKGOZ
   Nine Hatun Mh. 49. Sk. No:40
   Esenler, Istanbul TURKEY

6. Deniz Kuvvetleri Komutanligi
   Personel Daire Baskanligi
   Bakanliklar, Ankara TURKEY

7. Deniz Harp Okulu Komutanligi
   Kutuphane
   Tuzla, Istanbul TURKEY

8. Kara Harp Okulu Komutanligi
   Kutuphane
   Bakanliklar, Ankara TURKEY

9. Hava Harp Okulu Komutanligi
   Kutuphane
   Yesilyurt, Istanbul TURKEY

10. Middle East Technical University(ODTU)
    Kutuphane
    Balgat , Ankara TURKEY

11. Bilkent University
    Kutuphane
    Bilkent, Ankara Turkey

12. Bogazici Universitesi
    Kutuphane
    Bebek, Istanbul Turkey